



Intelligent Health Monitoring System Using Deep Learning

Ehsaan Ali (1780104)

Daim Attiq (1780144)

Supervised by: Syed Muhammad Usman

In partial fulfillment of requirement for the degree

Bachelor of Science (Software Engineering)

Shaheed Zulfikar Ali Bhutto Institute of Science and Technology

Islamabad, Pakistan

September, 2021

Dedication

First of all, we dedicate our project to the creator Allah Almighty and dedicate to whom the world owes its existence Muhammad (Peace Be Upon Him). Next big thanks go to Mr. Syed Muhammad Usman as our Supervisor for this project and the opportunity given to us for doing our final year project. We are thankful for his guidance in order to complete this project from the beginning until the end of this project. Last but not the least, with deepest gratitude we dedicate this project to our beloved parents, extremely dedicated and generous faculty of Computer Science Department and supportive friends, their prayers always pave the way to success for us.

Project Overview

Healthcare in Pakistan is one of the country's evolving sector. Improving health care services requires strengthening various aspects of healthcare services and it is only possible through a data-driven intelligent system. The problem with the existing healthcare monitoring dashboards is that they are limited because their User Interface (UI) is not interactive and data is not presented in an effective way. They also fail to provide the necessary information such as trend analysis and disease forecasting which is extremely useful in decision making. To overcome these limitations we have developed an intelligent dashboard that utilizes the data generated by the healthcare sector in improving healthcare services and tackle problems in overlooked health programs such as Tuberculosis program, Expanded Program of Immunization, and Lady Health Worker Program. To develop the proposed system we have followed Scrum methodology which is an agile framework for development of a software product. We have chosen it because it is flexible model and supports requirement change even after development is started. Moreover, our system also utilizes a large volume of data provided by the Sindh government related to the above-mentioned healthcare programs to perform disease forecasting using deep learning algorithm which will ultimately help in improving healthcare quality and decision making. While implementing deep learning, data preprocessing was performed to remove missing values, attribute selection was done using domain knowledge and then model building evaluation was performed. Linear Regression and Lasso Regression algorithms were used in this process. Our product is intended to be used by decision/policy makers of Sindh to assist them in monitoring of the healthcare programs and understanding the healthcare trends. The disease forecasting capability of our system has the ability to help decision makers to prepare for future risks and threats in advance.

Contents

Project Overview

List of Figures	v
-----------------	---

List of Tables	vi
----------------	----

1 Introduction	1
1.1 Product Purpose	1
1.2 Product Scope	2
1.2.1 Existing System Description	2
1.2.2 Future System Usage Analysis	4
1.3 Objectives	5
1.4 Problem Statement	5
1.5 Proposed Solution	5
1.6 Intended Market of Product	6
1.7 Intended Users of Product	6
1.7.1 Process Model Introduction	7
1.7.2 Justification	7
1.7.3 Steps of Process Model	7
2 SOFTWARE REQUIREMENTS SPECIFICATION	9
2.1 Introduction	9
2.1.1 Document Scope	9
2.1.2 Audience	9
2.2 Functional Requirements	10
2.2.1 Functional Requirements	10
2.3 Non-Functional Requirements	12
2.3.1 Software Quality Attributes	12
2.3.2 Other Non-Functional Requirements	13
2.4 Requirement Gathering Techniques Used	13
2.4.1 Brainstorming:	13
2.4.2 One-on-One Interview:	13
2.5 Time Frame	14
3 SOFTWARE PROJECT PLAN	15
3.1 Deliverables of the Project	15
3.2 Software Project Management Plan	16
3.2.1 Project Planning	16

3.2.2	Project Scheduling	19
3.2.3	Gantt Chart	19
3.2.4	Work Breakdown Structure	20
3.2.5	Critical Path Method	21
3.3	Managerial Process	21
3.3.1	Management Objectives and Priorities	22
3.3.2	Assumptions and Constraints	22
3.4	Project Risk Management	23
3.4.1	Risk Management Plan	24
3.4.2	Risk Management Activities	25
4	FUNCTIONAL ANALYSIS AND MODELING	27
4.1	Use Case Modeling	27
4.1.1	User Stories	27
4.1.2	Individual Actor Use Cases	29
4.1.3	Use Case Diagram	36
4.2	Functional Modeling	37
4.2.1	Entity Relationship Diagram	37
4.2.2	Relational Schema	38
4.2.3	Data Flow Diagram	39
5	SYSTEM DESIGN	41
5.1	Structure Diagrams	41
5.1.1	Class Diagram	41
5.1.2	Deployment Diagrams	42
5.2	Behavioral Diagrams	43
5.2.1	Activity Diagrams	43
5.2.2	Communication Diagrams	44
5.2.3	Sequence Diagrams	45
6	SYSTEM INTERFACE AND PHYSICAL DESIGN	55
6.1	System User Interfaces	55
6.1.1	Home Page	55
6.1.2	Home Page	56
6.1.3	Sign-In Interface	56
6.1.4	TB Section	57
6.1.5	EPI Section	57
6.1.6	LHW Section	58
6.1.7	LHW Evaluation: Area Wise	58
6.1.8	LHW Evaluation: Report Wise	59

6.1.9	Trends Analysis	59
6.1.10	Sign-Up Interface	60
6.1.11	Update Record Interface	60
6.1.12	Admin Table	61
6.1.13	User Table	61
6.1.14	TBP Table	62
6.1.15	EPI Table	62
6.1.16	LHWP Table	63
6.1.17	Record Table	63
6.1.18	Report Table	64
7	TEST PLAN	65
7.1	Objective of the Testing Phase	65
7.2	Levels of Tests for Testing Software	66
7.2.1	Unit Testing	66
7.2.2	Integration testing	66
7.2.3	System Testing	66
7.3	Test Management Process	67
7.3.1	Design the Test Strategy	67
7.3.2	Test Criteria	68
7.3.3	Resource Planning	68
7.3.4	Plan Test Environment	69
7.3.5	Schedule and Estimation	70
7.4	Test Cases With Description	71
7.5	Bugs Report	80
8	CONCLUSION	81
8.1	Conclusion	81
8.2	Future Work	82
	REFERENCES	82

List of Figures

1.1	System Diagram	6
1.2	Sprint Life-cycle	8
3.1	Gantt Chart	19
3.2	Work Breakdown Structure	20
3.3	Critical Path Diagram	21
4.1	Use Case Diagram	36
4.2	Entity Relationship Diagram	37
4.3	Relational Schema	38
4.4	DFD Level 0	39
4.5	DFD Level 1	39
4.6	DFD Level 2	40
5.1	Class Diagram	41
5.2	Deployment Diagram	42
5.3	Activity Diagram	43
5.4	Communication Diagram: Admin	44
5.5	Communication Diagram: User	44
5.6	Admin: Login	45
5.7	Admin: Add New User	46
5.8	Admin: Update Record	47
5.9	User: Login	48
5.10	User: Monitor Healthcare Program	49
5.11	User: Select District	50
5.12	User: Monitor Health Status	51
5.13	User: Generate Healthcare Report	52
5.14	User: Analyze Healthcare Trends	53
5.15	User: Disease Forecasting	54
6.1	Home Page	55
6.2	Home Page	56
6.3	Sign-In Interface	56
6.4	TB Section	57
6.5	EPI Section	57
6.6	LHW Section	58
6.7	LHW Evaluation: Area Wise	58
6.8	LHW Evaluation: Report Wise	59

6.9 Trends Analysis	59
6.10 Sign-Up Interface	60
6.11 Update Record Interface	60
6.12 Admin Table	61
6.13 User Table	61
6.14 TBP Table	62
6.15 EPI Table	62
6.16 LHWP Table	63
6.17 Record Table	63
6.18 Report Table	64

List of Tables

1.1	Abbreviations	2
1.2	Comparison of Existing Healthcare Systems	3
1.3	Literature Review	4
1.4	Future System Usage Analysis	5
2.1	Time Frame	14
3.1	Project Planning	17
3.2	Milestones Plan	17
3.3	Documentation Plan	18
3.4	Resource Plan	18
3.5	Project Scheduling	19
3.6	Probability of Risks	23
3.7	Risk Management Plan	24
3.8	Roles and Responsibilities	24
3.9	Rating Risk Impact	25
3.10	Prioritizing All Risks w.r.t Impact	26
7.1	Resource Planning	68

Chapter 1

Introduction

The healthcare sector is emerging through the use of advanced analytic and data science technologies. Current investment in the infrastructure and IT development is a reflection of this emerging trend. In spite of this trend, countries are slow in providing healthcare services due to lack of data-driven analytical support. Information management in healthcare is a major task in the healthcare profession and currently there is a need to focus our attention on healthcare sector [1]. This demand for intelligent healthcare monitoring services is the reason behind this project. Another opportunity with the presence of predictive analytics is that it can be utilized to save time and help in decision making. Based on the gathered data and past trends, the system predicts that there is a risk that the number of patients of TB is expected to increase in a specific city, then the stakeholders will be able to prepare for that in advance.

The predictive analytics feature of the system also helps in allocating staff in the required city/region. Using the system the stakeholders are able to predict the increase in the ratio of patients over a given period allowing them to allocate specialist doctors or lady health workers based on those numbers. Health care in Pakistan is identified as one of the country's neglected sector. Improving health care services requires strengthening various aspects of healthcare services and it is only possible through a data-driven intelligent system [2].

1.1 Product Purpose

Pakistan's healthcare sector generates large amount of data in a variety of forms. It is difficult to utilize this large data for meaningful knowledge discovery. The purpose behind developing an intelligent healthcare system is to allow stakeholders and decision makers to monitor the following healthcare programs: Tuberculosis program, Expanded Program of Immunization and Lady Health Worker Program to assist the healthcare sector in improving healthcare services.

Our system is designed to handle large amount of healthcare data and present it in a meaningful way. The system is also be useful for understanding the current trends of healthcare programs. Those trends help the stakeholders in decision making and also in the utilization of resources. Another major purpose of our product is to assist decision makers and stakeholders in informed decision making by performing disease forecasting

allowing them to prepare for future threats and risks in advance.

1.2 Product Scope

The scope of the system is to enable the user to perform intelligent monitoring of Tuberculosis program (TB), Expanded Program of Immunization (EPI), and Lady Health Worker Program (LHWP) in desired cities of the province based on gathered data. Our intelligent system also evaluates the performance of Lady Health Worker on the basis of data provided by Govt. of Sindh. The system also enables the user to understand the trends in the above mentioned healthcare programs and perform disease forecasting through deep learning.

Table 1.1: Abbreviations

Serial #	Abbreviation	Definition
1	TBP	Tuberculosis Program
2	EPI	Expanded Program of Immunization
3	LHWP	Lady Health Worker Program
4	WHO	World Health Organization

1.2.1 Existing System Description

City Health Dashboard – US [3] was launched initially with data on over 35 diseases and was available for 500 cities across the U.S. Since then, the dashboard has added 29 more diseases and also improved data visualizations. Today, the dashboard covers over 750 cities across the U.S. As shown in table 1.2, it consist of features like Health Monitoring, Display Death Ratio, Density Map, Comparative Analysis of Cities, Disease Comparison, and Report Generation. This dashboard is limited in the sense that it does not provide its user with exact values e.g. exact number of death due to a disease. It only displays average deaths per 100,000 patients.

Public Health Dashboard – UK [4] is a UK based dashboard managed by the Government of the England. This dashboard allows its user to monitor health status area wise, region wise and city wise. This dashboard contains health data of the complete country from the year 2006. As shown in table 1.2, its features include Health Monitoring, Display Death Ratio, Density Map, Comparative Analysis of Cities, Report Generation, and Trends Analysis. This dashboard has everything feature wise, only issue with this dashboard is that it has an unattractive UI because of which it is difficult for a new user to understand the information displayed.

WHO’s Dashboard [5] contains health data of more than 190 countries and more than 30 health care programs. This dashboard allows its users to explore healthcare

programs and see how much a country is suffering from a particular disease. As shown in table 1.2, it comprise of features like Health Monitoring, Display Death Ratio, and Report Generation. The WHO’s dashboard does not incorporate a map that shows health status of a country. A map should be available that facilitates in understanding which part of the country is less affected by the disease and which is more.

Health Dashboard – Canada [6] is the Government of Canada’s dashboard that is open for public. The goal of this dashboard is to make it easier for Canadians to understand the healthcare status of the country. People can also learn how the government is working to improve the healthcare services in the country. It covers more than 40 healthcare programs. As shown in table 1.2, it has features of Health Monitoring, Comparative Analysis of Cities, and Report Generation. The issues with this dashboard is that its UI is extremely unattractive and old making it displeasing to the eye and difficult to use.

Health Dashboard – Bangladesh [7] was developed recently in the 2020 under the supervisions of Rockefeller foundation which is a private NGO working for the betterment of healthcare in Bangladesh. It covers 22 healthcare programs with major focus on Pneumonia, Diarrhea and Infant Mortality. As shown in table 1.2, it incorporates features like Health Monitoring, Display Death Ratio, and Trends Analysis. This dashboard is limited in the sense that it does not provide its user with exact values e.g. exact number of death due to a disease. It only displays average deaths per 100,000 patients.

Table 1.2: Comparison of Existing Healthcare Systems

Feature	City Health Dashboard [3]	Public Health Dashboard-UK [4]	WHO’s Dashboard [5]	Health Dashboard-Canada [6]	Health Dashboard-Bangladesh [7]
Health Monitoring	✓	✓	✓	✓	✓
Display Death Ratio	✓	✗	✓	✗	✓
Density Map	✓	✓	✗	✗	✗
Comparative Analysis of Cities	✓	✓	✗	✓	✗
Disease Comparison	✓	✗	✗	✗	✗
Report Generation	✓	✓	✓	✓	✗
Trends Analysis	✗	✓	✗	✗	✓

Literature Review:

A data-driven framework for introducing predictive analytics into expanded program on immunization in Pakistan[8] is a research article in which the defaulter prediction model has been proposed and a number of classifiers like Decision Tree, Multilayer Perceptron, Support Vector Machine, and Naive Bayes have been applied on the PDHS dataset. Table 1.3 shows that the proposed framework is able to achieve 94 percent accuracy using Naive Bayes classification technique.

The findings of the research also has some limitations. Firstly, they have only used a very limited number of demographic attributes which were present in the Pakistan Demographic and Health Survey (PDHS) dataset. Other information-rich attributes available in other datasets might show contradicting results. Secondly, there method of labeling the different categories of defaulters does not include children who get vaccinated during different immunization campaigns such as the polio campaign.

A machine learning-based framework for Predicting Treatment Failure in tuberculosis: A case study of six countries[9] is a research article in which the machine learning approach has been proposed for predicting treatment failure in TB. The data related to TB of six different countries was used and a number of classifiers like ANN, kNN, Support Vector Machine, RF and J48 have been applied on the NIAID dataset. Table 1.3 shows that the proposed framework is able to achieve 78 percent accuracy using ANN classification technique. The approach has a number of applications in the health care sector. It can be used in the analysis and prediction of treatment failure. The attribute suggested by this approach can be utilized to predict the outcome of patient's treatment.

Table 1.3: Literature Review

Authors	Sadaf et al. [8]	Asad et al. [9]
Program	Expanded Program of Immunization	TB Program
Dataset Source	Pakistan Demographic and Health	National Institute of Allergy and Infectious Diseases
Approach	Machine Learning	Machine Learning
Algorithm	Naïve Bayes	ANN

1.2.2 Future System Usage Analysis

The existing dashboards are only used for the purpose of health monitoring only and have limited set of features. But our Intelligent Dashboard will be used by policy makers to critically monitor the health status of the province using density maps, tables, pie charts, and graphs. As shown in table 1.4, our interactive dashboard will also be used to evaluate the performance of Lady Health Workers and perform trends analysis as well. Disease forecasting will facilitate the policy makers in decision making to prepare for future threats.

Table 1.4: Future System Usage Analysis

Feature	City Health Dashboard [3]	Public Health Dashboard-UK [4]	WHO's Dashboard [5]	Health Dashboard-Canada[6]	Health Dashboard-Bangladesh[6]	Proposed System
Health Monitoring	✓	✓	✓	✓	✓	✓
Display Death Ratio	✓	✗	✓	✗	✓	✓
Density Map	✓	✓	✗	✗	✗	✓
Comparative Analysis of Cities	✓	✓	✗	✓	✗	✓
Disease Comparison	✓	✗	✗	✗	✗	✓
Report Generation	✓	✓	✓	✓	✗	✓
Trends Analysis	✗	✓	✗	✗	✓	✓
Disease Forecasting	✗	✗	✗	✗	✗	✓
Performance Evaluation of LHW	✗	✗	✗	✗	✗	✓

1.3 Objectives

Our objectives are to develop a system that can:

- Monitor the mentioned healthcare programs and assist the healthcare sector in improving healthcare services.
- Handle large amount of healthcare data and present it in a meaningful way.
- Assist in informed decision making by providing data insights and predicting future risks.

1.4 Problem Statement

Existing healthcare dashboards are limited because their UI is not interactive and data is not presented in an effective way. They also fail to provide the necessary information such as trend analysis and disease forecasting which is extremely useful in decision making.

1.5 Proposed Solution

The proposed system is a web-based dashboard that will utilize the data generated by the healthcare sector in improving healthcare services and tackle problems in overlooked

health programs such as Tuberculosis program, Expanded Program of Immunization, and Lady Health Worker Program. The system will also provide details regarding the Lady Health Workers working in a particular city and their performance. Moreover, the proposed system will also utilize a large volume of data provided by the Sindh government related to the above-mentioned healthcare programs to perform predictive analysis using deep learning algorithm which will ultimately help in improving healthcare quality and decision making. Figure 1.1 shows interaction between multiple components of the system and how the data is being transferred between them.

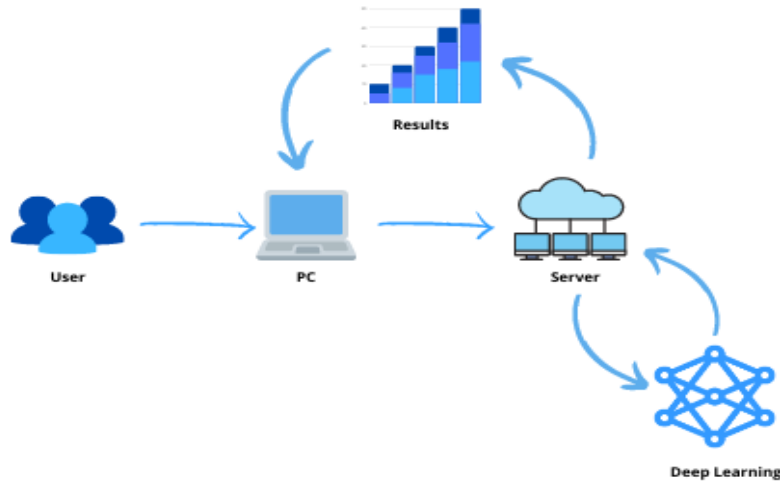


Figure 1.1: System Diagram

1.6 Intended Market of Product

Our product is intended to be used by healthcare sector of Sindh helping them in improving healthcare services in the province. Our system will utilize the data provided by govt. of Sindh to assist their healthcare sector in decision making.

1.7 Intended Users of Product

Our product will be used by decision/policy makers of Sindh to assist them in monitoring of the healthcare programs and understanding the healthcare trends. The disease forecasting capability of our system will help them to prepare for future risks and threats in advance.

1.7.1 Process Model Introduction

To develop the proposed system we will follow Scrum methodology which is an agile framework for development of a software product. It is one of the approaches that is used primarily for development of software projects with the goal of delivering new functionality or capability every sprint. A sprint is a time boxed period and is usually of 2 to 4 weeks but can be of one week as well. Scrum is the most popular agile methodology and is being used widely by software development teams to develop high quality software faster [10].

1.7.2 Justification

The Scrum methodology follows the principles and values of agile which is a combination of incremental and iterative process models. We have chosen it because it is flexible model and supports requirement change even after development is started. It also supports incremental delivery after each time span and it is also suitable for projects involving less team members. Further benefits that are received by adopting Scrum methodology include higher productivity, better quality products, improved stakeholder satisfaction, and reduced time to market [11].

1.7.3 Steps of Process Model

As shown in figure 1.2, the scrum life-cycle process consist of following phases:

1. **Product Backlog Creation:** First we will create a product backlog, which is a document that outlines every requirement of a product/system. The product backlog is a type of a to-do list consisting of features/items which are ordered in terms of business value.
2. **Product Backlog Refinement:** During this step we will review and revise the product backlog items. It is an ongoing process in which the development team and the product owner negotiate on the details of product backlog items.
3. **Sprint Planning:** In planning phase we will decide what are we going to work on and how we will do it. For this purpose a sprint planning meeting will take place.
4. **Sprint Backlog Creation:** Next we will create a sprint backlog and each sprint will be assigned specific items/features which we will be delivering at the end of the sprint. For our project each sprint will be of 4 weeks.
5. **Sprint Implementation:** After creating the sprint backlog, each member of the team will be assigned tasks according to the sprint backlog. A scrum meeting will be held on daily basis for about 15 minutes in which group members will report what they

yesterday, and commit what will they do today, and ask questions related to any problems that they are facing.

6. **Sprint Review:** We will conduct a sprint review meeting at the end of the sprint to ensure that the delivered increment met the requirements and revisit the product backlog on the basis of the received feedback.

7. **Sprint Retrospect:** A sprint retrospective meeting will be held after we have reviewed the sprint. In this 90 minute meeting, the team will identify the items/features that we fail to deliver during the sprint and work on its solution as well.

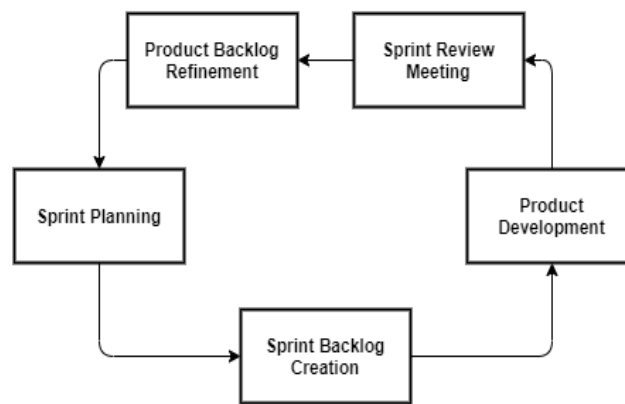


Figure 1.2: Sprint Life-cycle

Chapter 2

SOFTWARE REQUIREMENTS SPECIFICATION

2.1 Introduction

The following SRS contains the detail perspective of the system from multiple stakeholders. It also provides a general description, including characteristics of the users of this project, the product's hardware, and the functional and non-functional requirements of the product. The purpose of the SRS document is to collect and analyze ideas that have come up to define the system and its requirements w.r.t consumers of the product.

This SRS will also provide a broad overview of our software product, its parameters and goals to be achieved. This document describes the projects target audience and its hardware and software requirements. It defines how our client, team and audience see the product and its functionality. It also assists the designer and developer in SDLC processes.

2.1.1 Document Scope

This SRS is aimed at specifying requirements of software to be developed. The purpose of this document is to present a detailed description of the system to be developed. It will explain what the system will do, the purpose and features of the system, how the system will react under specific conditions, and the constraints under which it must operate.

2.1.2 Audience

This document is intended for both the stakeholders of the project and the developers of the system but primarily it is written for the developers and describes the details of the functionality of the product in technical terms.

Project testers can also use this document to assist them in testing phase as some bugs are easier to find with the help of the requirements document. This way testing will become more organized and less time consuming.

2.2 Functional Requirements

2.2.1 Functional Requirements

1. Register/Sign-In:

1.1. The system shall allow the admin to register a new user.

1.2. The system should allow a user to sign-in using a username and a password.

2. Healthcare Programs:

The system shall allow the user to selected between healthcare programs e.g. TB, EPI, and LHW

3. Monitoring:

3.1 The system shall use tables, graphs, and charts for monitoring each healthcare programs.

3.2 The system shall allow the user to select a city from drop down menu after entering a specific healthcare category.

3.3 In TB Program the system shall display following information:

3.3.1 Total no. of Cases

3.3.2 No. of Patients whose Treatment Completed

3.3.3 No. of Patients whose Treatment Failed

3.3.4 No. of Deaths

3.3.5 No. of Patients that lost to follow up

3.4 In EPI the system shall display following information:

3.4.1 No. of Births

3.4.2 No. of children's fully immunized

3.4.3 No. of Infants Whose Immunization has started

3.5 In LHW Program the system shall display following info:

3.5.1 No. of Infant Mortality

3.5.2 No. of Neo-natal Death

3.5.3 No. of Still Birth

3.6 The system shall allow the user to evaluate the performance of a LHW by uploading her reports.

4. Density Maps:

4.1 Density maps should be available to assist in the monitoring process for each healthcare programs.

4.2 The system shall allow the user to see health status of a specific city by interacting with the density map.

5. Comparative Analysis:

5.1 The system shall allow the user to compare multiple cities for status comparison by selecting cities from the drop down menu.

5.2 The system shall allow the user to compare programs/diseases for status comparison by selecting programs/diseases from the drop down menu.

6. Report Generation:

6.1 The system shall allow the user to download healthcare reports in the .csv format.

6.2 The system shall allow the user to choose time span of report to be downloaded e.g. Annual Report, Quarterly report, Monthly Report or Custom report.

7. Trends Analysis:

The system should display current disease trends on the basis of existed data in the form of graphs.

8. Disease Forecasting:

The system should also be able to perform disease forecasting on the basis of existing healthcare data.

9. Update Record:

The system shall allow the user to upload files and reports to keep the healthcare data updated.

2.3 Non-Functional Requirements

2.3.1 Software Quality Attributes

1. Usability:

1.1 The system shall provide a uniform feel and look between the web pages of the dashboard.

1.2 The system shall use standards icons and toolbars to improve usability.

2. Reliability:

The system shall provide for replication of databases to off-site storage locations.

3. Availability:

The system shall be provided with a stable internet connection so that it is available all the time.

4. Performance:

4.1 The product is a web based product and has to be run from a web server.

4.2 The product shall take initial loading time depending on the strength of the provided internet connection.

4.3 The performance shall also depend on hardware on which the system is accessed by the user.

5. Security:

5.1 The system shall never display the user's password. It shall always be represented with special characters.

5.2 The system's back end servers shall also never display a user's password. The user's password shall be reset but never shown.

5.3 The system's back end servers shall only be accessible by an authenticated administrators.

5.4 The system's back end databases shall be encrypted to avoid hacking.

2.3.2 Other Non-Functional Requirements

1. Hardware Interface:

Since the application is web based and runs over the internet, so all the hardware required to connect with the internet will be hardware interface e.g. Modem, WAN/LAN, Ethernet Cable.

2. User Interface:

The user interface of the software should be compatible to any web browser such as Internet Explorer, Chrome, Mozilla by which user will access the dashboard.

3. Legal, Copyright, and Other Notices:

The system shall display the disclaimers, copyright, and trademark information as well.

2.4 Requirement Gathering Techniques Used

2.4.1 Brainstorming:

We have used the method of brainstorming for identifying and generating requirements for our intelligent healthcare dashboard. In the brainstorming session team sits together and generate as many ideas as possible and share with each other. Brainstorming is different from a conventional meeting and encourages idea sharing without the fear of judgment. We conducted 30 to 60 minutes brainstorming sessions for identifying requirements for our system and to understand how we will approach those requirements. [12]

2.4.2 One-on-One Interview:

A one on one interview with the representative of Sindh's healthcare sector was conducted to identify and refine the requirements. In this type of methodology it is easier to ask follow-up questions in order to get more detailed and accurate information. We asked open ended as well as close ended questions to get the maximum out of the interviewee. Open ended questions were helpful in identifying requirements along with its details. Close ended questions were helpful in gathering more requirements in less time. [12]

2.5 Time Frame

Time frame / duration of requirement phase is given below in table 2.1.

Table 2.1: Time Frame

Sr no.	Phase	Duration
1	Inception	4 days
2	Elicitation	7 days
3	Elaboration	3 days
4	Negotiation	7 days
5	Specification	4 days
6	Validation	3 days
7	Requirements Management	7 days

Chapter 3

SOFTWARE PROJECT PLAN

3.1 Deliverables of the Project

As we are using scrum methodology which is an agile framework for developing software projects, it focuses on flexibility, continuous improvement, and high quality results. It aims to be measurable by using deliverables as a mean to track the progress of the project [1]. Deliverable are usually measurable outputs that are created as a result of work done during the course of the project. So keeping in the scrum methodology, following are the deliverable of our Intelligent Health Monitoring Dashboard:

1. SRS Document: The SRS document of the project will provides a general description, including characteristics of the users of this project, functional and non-functional requirements of the product. The purpose of the SRS document is to collect and analyze ideas that have come up to define the system and its requirements w.r.t consumers of the product.

2. Design Document: The software design document of our system will provide a high level description of the systems design that will allow the development team to proceed with a clear understanding of what is to be built.

3. Project Plan: A project plan will be helpful in understanding how we will proceed the project along with identifying and prioritizing the tasks and resources required to create the structure for our project.

4. Product Backlog: We will also deliver a product backlog which is a document that outlines every requirement of a product/system. The product backlog will contain a full list of what is needed to be done to complete your project and will be ordered by priority.

5. Sprint Backlog: This document will contain user stories, goals, and tasks associated with the respective sprint and will be delivered after the product backlog.

6. Interface Mockups: Interface mockups will also be provided to the stakeholders which will show what the final interface of the system will look like so that the stakeholders can recommend changes if any.

7. Prototypes: We will be delivering system prototypes as well from the initial

requirements to obtain early feedback from the stakeholders.

8. Database: A database schema of the database of our project will be developed which will represent the logical configuration of our relational database, and it will be delivered to stakeholders to get feedback.

9. Complete User Interface: After getting feedback from the stakeholders using interface mockups, the complete interface of the system will be developed which will be ready to be integrated with the database.

10. Integrated System: After finalizing the UI, the system will be integrated with the database which will be developed using Postgresql. System will be able to perform disease forecasting which will be possible through deep learning. System will be fully functional at this point and will give dynamic responses to the user using its database.

11. Test Report: Before finalizing the final product, the system will go through the testing phase. A test report containing the test results will be delivered and system will go for review if required.

12. Final Product: After removing all the bugs, errors or problems identified during the testing phase, a final product as per requirements of the customers/stakeholders will be deployed to web after purchasing the domain

3.2 Software Project Management Plan

3.2.1 Project Planning

Poor planning is constantly identified as one of the top reasons for project failure. Without developing a suitable project plan at the very beginning, determining the scope of a project becomes practically impossible. Without a project plan it is difficult to keep track of resources, tools and techniques that are required to accomplish the goals. Therefore a good project plan has to be developed in order to successfully complete the project within the given schedule and budget.

In order to deliver the above mentioned project deliverables we require sufficient human resources, time, and budget. As we are two group members who will be working on the project and we need to deliver the project in approximately 10 months, so we have to make a plan accordingly. Our Project Plan is given below in table 3.1, which shows all the tools, technologies and other resources that will be utilized for each of the deliverable, followed by Documentation Plan, Resource Plan and a Gantt Chart of our project.

Table 3.1: Project Planning

<i>SR no.</i>	<i>Deliverable</i>	<i>Ehsaan Ali</i>	<i>Daim Attiq</i>	<i>Tools</i>	<i>Technology</i>	<i>Budget</i>	<i>Time</i>
1	SRS Document	✓	✓	MS Word	—	—	2 Weeks
2	Design Document	—	✓	MS Word	—	—	1 Weeks
3	Project Plan	✓	✓	MS Word, MS Project	—	—	2 Weeks
4	Product Backlog	✓	✓	MS Word/MS Excel	—	—	1 Weeks
5	Sprint Backlog	✓	—	MS Word, MS Excel	—	—	1 Weeks
6	Interface Mockups	✓		Draw.io, Visual Paradigm	—	—	2 Weeks
7	Prototypes	✓	✓	Visual Studio Code, Anaconda	Django, HTML, CSS	—	2 Weeks
8	Database	—	✓	PGAdmin, Draw.io	Postgresql	—	2 Weeks
9	Complete User Interface	✓	✓	Visual Studio Code	Django, HTML, CSS	—	10 Weeks
10	Integrated System	✓	✓	Visual Studio Code, Anaconda, PG Admin, Postgresql	Django, HTML, CSS, Postgresql, Deep Learning	25,000 (GPU for Deep Learning)	12 Weeks
11	Test Report	✓	✓	MS Word, MS Excel	—		2 Weeks
12	Final Product	✓	✓	Visual Studio Code, Anaconda, PG Admin, Postgresql	Django, HTML, CSS, Postgresql, Deep Learning	20,000 (Domain and Cloud Hosting)	4 Weeks

3.2.1.1 Milestones Plan

Table 3.2: Milestones Plan

SR No.	Milestone	Milestone Goal	Duration
1	Requirement Engineering	Gathering requirements from client, Understanding the requirements, Negotiating requirements (If Needed), Validating the requirements.	2 Weeks
2	Sign-Up/Sign-In	Sign-Up and Sign-In pages.	1 Week
3	Home Page	Home Page of the dash board allowing the user to navigate between the healthcare programs.	1 Week
4	Monitoring	This section will allow the user to monitor the status of the TB Program, Immunization Program and Lady Health Worker Program in a district or city.	2 Weeks
5	Density Map	An interactive map that will assist the user in the monitoring process.	2 Weeks
6	Report Generation	An interactive interface that will allow the user to download healthcare reports in the .csv format e.g. Annual Report, Quarterly report, Monthly Report or Custom report.	1 Week
7	Trend Analysis	Charts and Graphs will display the current healthcare trends.	2 Weeks
8	Disease Forecasting	This section will perform disease forecasting on the basis of existing healthcare data.	6 Weeks
9	Testing	Error free product and a Project Report.	2 Weeks

3.2.1.2 Documentation Plan

Table 3.3: Documentation Plan

SR no.	Document Name	Documenter Name	Duration
1	Project Proposal	Ehsaan, Daim	1 Week
2	Chap 1 & 2	Ehsaan, Daim	3 Weeks
3	Chap 3 & 4	Ehsaan, Daim	3 Weeks
4	Chap 5 & 6	Ehsaan, Daim	3 Weeks
5	Chap 7 & 8	Ehsaan, Daim	3 Weeks
6	User Manual	Daim	1 Week

3.2.1.3 Resource Plan

Table 3.4: Resource Plan

<i>SR no.</i>	<i>Resource Name</i>	<i>Task</i>	<i>Budget</i>	Time Required
1	Ehsaan, Daim	Requirement Gathering	-	1 Weeks
2	Ehsaan, Daim	Develop SRS Document	-	2 Weeks
3	Daim	Develop Design Document	-	1 Weeks
4	Ehsaan, Daim	Develop Project Plan	-	2 Weeks
5	Ehsaan, Daim	Develop Product Backlog	-	1 Weeks
6	Ehsaan	Develop Interface Mockups	-	2 Weeks
7	Ehsaan, Daim	Build Prototypes	-	2 Weeks
8	Daim	Develop Database	-	2 Weeks
9	Ehsaan, Daim	Complete User Interface	-	10 Weeks
10	Ehsaan, Daim	Implement Deep Learning	25,000	12 Weeks
11	Ehsaan, Daim	Perform Testing	-	2 Weeks
12	Ehsaan, Daim	Develop Test Report	-	1 Weeks
13	Daim	Develop User Manual	-	1 Weeks
14	Ehsaan	Deploy System On Web	20,000	1 Weeks

3.2.2 Project Scheduling

Table 3.5: Project Scheduling

<i>SR. no</i>	<i>Deliverable</i>	<i>Start Date</i>	<i>End Date</i>
1	SRS Document	Oct/05/2020	Oct/19/2020
2	Design Document	Oct/20/2020	Oct/27/2020
3	Project Plan	Oct/28/2020	Nov/04/2020
4	Product Backlog	Nov/05/2020	Nov/12/2020
5	Sprint Backlog	Nov/13/2020	Nov/20/2020
6	Interface Mockups	Nov/21/2020	Dec/05/2020
7	Prototypes	Nov/20/2020	Dec/05/2020
8	Database	Nov/20/2020	Dec/05/2020
9	Complete User Interface	Nov/20/2020	Feb/05/2021
10	Integrated System	Feb/05/2021	May/13/2021
11	Test Report	May/14/2021	May/28/2021
12	Final Product	May/29/2021	June/26/2021

3.2.3 Gantt Chart

Task	Oct	Nov	Dec	Jan	Feb	Mar	Apr	May	Jun
SRS Document									
Design Document									
Project Plan									
Product Backlog									
Sprint Backlog									
Interface Mockups									
Prototypes									
Database Schema									
Complete User Interface									
Integrated System									
Test Report									
Final Product									

Figure 3.1: Gantt Chart

3.2.4 Work Breakdown Structure

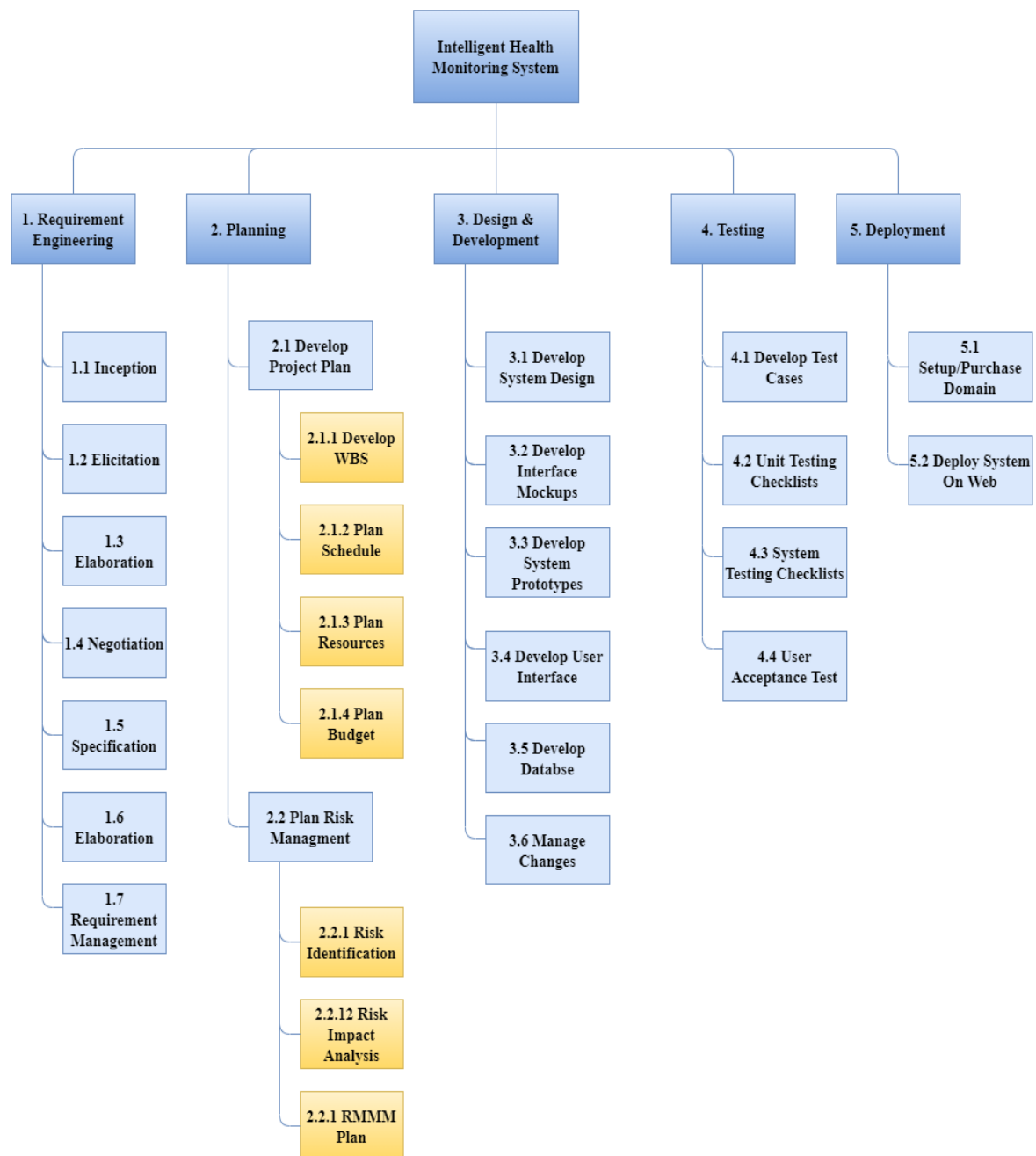


Figure 3.2: Work Breakdown Structure

3.2.5 Critical Path Method

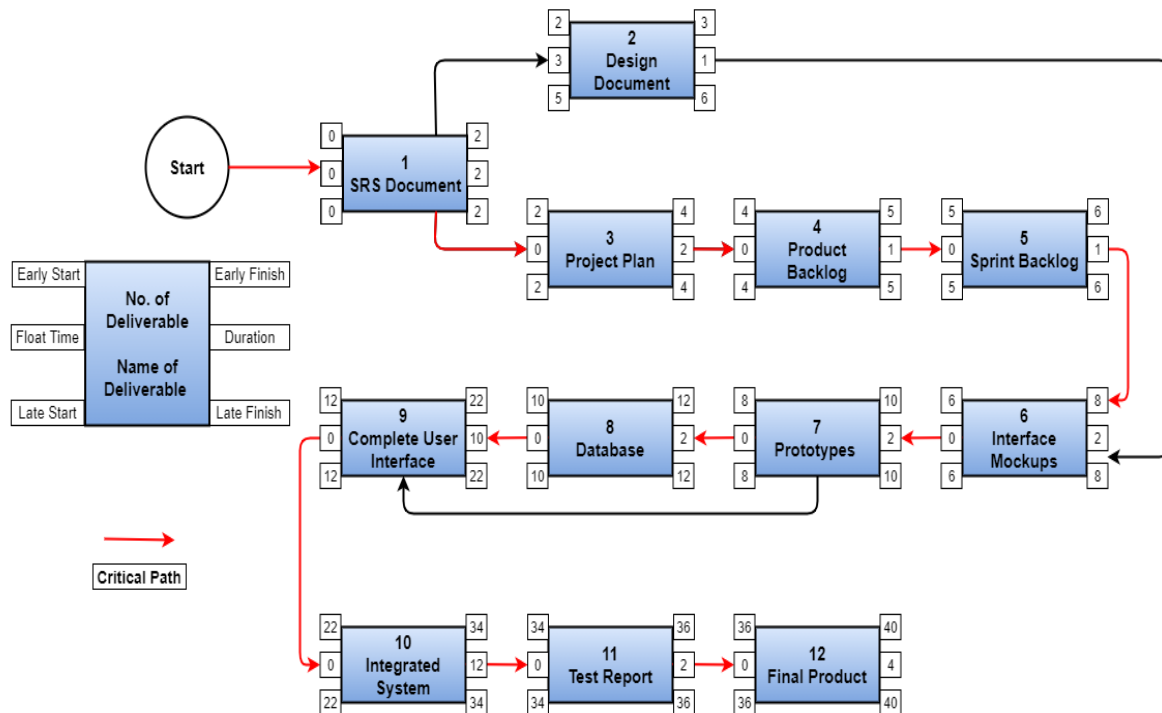


Figure 3.3: Critical Path Diagram

3.3 Managerial Process

Project management is considered to be a key factor in the success of a software project that is why a project manager needs to have all the sufficient knowledge along with enough managerial experience to efficiently carry out all the managerial process.

1. Planning Planning is the first process in project management, It involves defining the initial scope of the project, allocating of all necessary resources, breaking larger deliverables into smaller ones, Identifying tasks that are necessary to produce the deliverables, determine the duration of the tasks, identifying all the possible risks along with their impact and developing a plan to mitigate or manage those risks.

2. Execution After the planning phase, now it is important to successfully execute all the tasks identified during the planning phase. In this process the project manager has to participate and supervise the successful execution of each tasks of the project.

3. Monitor and Control During the execution phase, it is important that all aspects of the project tasks are being monitored so that project remains on track. Project manager has to monitor and control changes in scope, time and budget so that they do not contradict customer requirement. Also, monitor and control any risk that occurs during the project, so that it does not affects the project.

4. Closing Making sure that the project is delivered on time and project goals/objectives have been successfully achieved is also a part of managerial process. This process also emphasis on making sure that the project deliverables have been completed as planned.

3.3.1 Management Objectives and Priorities

Objectives:

1. The successful implementation of the tasks and sub-tasks of the project.
2. Efficient communication between all the members of the project team/teams.
3. Achieve all the project goals within-in the given constraints.
4. Optimization of all the resources allocated to the project.
5. Deliver a project that is according to the requirements and needs of the client/customer.

Priorities:

1. Make sure that all the quality objectives are met throughout the project.
2. The project deliverables must be completed as planned.
3. The project must be delivered on time with-in the allocated budget.
4. The project should not fall behind or exceeds the scope of the project.
5. Track all milestones, deliverables, and change request which occur during the project.

3.3.2 Assumptions and Constraints

Assumptions:

A few assumptions that were made from the project management perspective are as follows:

1. We have all the resources required to develop and deliver the project on time.
2. Team members have all the necessary skills and knowledge to develop a web based solution.
3. Two group members will be enough to develop and deploy the system on time.
4. The scope of the project will not change throughout the project.
5. Cost for purchasing the domain and cloud hosting will not change during the

project.

Constraints: In our web based project we had the choice to use PHP to integrate front end with backend and we had all the knowledge, skills and experience for it. But as we are implementing deep learning in our project, so it is a difficult and complex task to implement a PHP based project using deep learning. So, we have opt for django framework instead of PHP which is a reliable but new technology and it will take our extra time to learn this new technology.

3.4 Project Risk Management

Project risk management is the process of identifying risks that may occur during the life cycle of the project and then making a plan on how to respond to those so that the project remains on track and meet its desired goal. A risk is anything that can impacts or in other words affects the project's scope, timeline, or budget. As we are developing a web based system using a new technology known as Django which is a python based framework, so there is a risk that some hurdles may arise during the lifecycle of the project.

Some major risks, along with their occurrence probability are shown below in the table 3.6.

Table 3.6: Probability of Risks

<i>Sr. no</i>	<i>Risk</i>	<i>Category</i>	<i>Probability</i>
1	Failed to meet the requirements.	Project Risk	30 Percent
2	Both group members have sufficient development skills but lack experience.	Known Risk	30 Percent
3	Insufficient training to use latest tools	Known Risk	20 Percent
4	Non responsive/unsupportable software	Technical Risk	20 Percent
5	Change in requirements may occur.	Project Risk	40 Percent
6	Mastering Django which is a new technology may require extra time.	Technical Risk	10 Percent
7	There is a possibility that high chart & map API may not support Django framework.	Technical Risk	5 Percent
8	Shortage of financial resources	Business Risk	10 Percent
9	Less reuse than planned	Project Risk	40 Percent
10	Possible minor budget over-run	Business Risks	20 Percent

3.4.1 Risk Management Plan

Table 3.7: Risk Management Plan

<i>Sr. no</i>	<i>Risk</i>	<i>Category</i>	<i>Probability</i>	Severity	Impact Value	RMMM
1	Failed to meet the requirements.	Project Risk	30 Percent	Catastrophic	1	Risk Monitoring
2	Both group members have sufficient development skills but lack experience.	Known Risk	30 Percent	Critical	2	Risk Mitigation
3	Insufficient training to use latest tools	Known Risk	20 Percent	Marginal	3	Risk Mitigation
4	Non responsive/unsupportable software	Technical Risk	20 Percent	Catastrophic	1	Risk Monitoring
5	Change in requirements may occur.	Project Risk	40 Percent	Critical	2	Risk Management
6	Mastering Django which is a new technology may require extra time.	Technical Risk	10 Percent	Critical	2	Risk Monitoring
7	There is a possibility that high chart & map API may not support Django framework.	Technical Risk	5 Percent	Critical	2	Risk Monitoring
8	Shortage of financial resources	Business Risk	10 Percent	Critical	2	Risk Management
9	Less reuse than planned	Project Risk	40 Percent	Critical	2	Risk Management
10	Possible minor budget over-run	Business Risks	20 Percent	Negligible	4	Risk Management

3.4.1.1 Roles and Responsibilities

Table 3.8: Roles and Responsibilities

Resource Name	Title	Email	Project Roles	Responsibilities
Ehsaan Ali	Group Leader & Member	1780104@szabist-isb.pk	Requirement Engineer, Designer, Developer & Tester	Requirement Gathering, Develop SRS Document, Develop Project Plan, Develop Interface Mockups, Complete User Interface, Implement Deep Learning, Deploy System On Web
Daim Attiq	Group Member	1780144@szabist-isb.pk	Requirement Engineer, Designer, Developer & Tester	Requirement Gathering, Develop SRS Document, Develop Design Document, Develop Project Plan, Develop Database, Complete User Interface, Implement Deep Learning, Perform Testing

3.4.2 Risk Management Activities

3.4.2.1 Rating Risk Impact :

Table 3.9: Rating Risk Impact

<i>Sr. no</i>	<i>Risk</i>	<i>Category</i>	<i>Severity</i>	Impact/Affect
1	Failed to meet the requirements.	Project Risk	Catastrophic	Project Success and Schedule.
2	Non responsive/unsupportable software	Technical Risk	Catastrophic	Project Success and Schedule.
3	Change in requirements may occur.	Project Risk	Critical	Project Scope, Schedule & Cost.
4	Mastering Django which is a new technology may require extra time.	Technical Risk	Critical	Project Schedule.
5	There is a possibility that high chart & map API may not support Django framework.	Technical Risk	Critical	Project Scope, & Cost.
6	Both group members have sufficient development skills but lack experience.	Known Risk	Critical	Project Schedule & Cost.
7	Shortage of financial resources	Business Risk	Critical	Project Efficiency and Schedule.
8	Less reuse than planned	Project Risk	Critical	Project Schedule and Cost.
9	Insufficient training to use latest tools	Known Risk	Marginal	Project Schedule.
10	Possible minor budget over-run	Business Risks	Negligible	Project Success.

3.4.2.2 Risk Monitoring and Control .

Risk monitoring and control is the process of identifying, analyzing, and planning for newly discovered risks and managing identified risks. Throughout the process, we have to track identified risks, reveal new risks, implement risk response plans, and measure the risk response plans effectiveness. The key point is throughout this phase constant monitoring and due diligence is key to the success. The inputs to Risk Monitoring and Control are:

Risk Management Plan: The Risk Management Plan is details how to approach and manage project risk. The plan describes the how and when for monitoring risks. Additionally the Risk Management Plan provides guidance around budgeting and timing for risk-related activities, thresholds, reporting formats, and tracking.

Approved Change Requests: Approved change requests are the necessary adjustments to work methods, contracts, project scope, and project schedule. Changes can impact existing risk and give rise to new risk. Approved change requests are need to be reviews from the perspective of whether they will affect risk ratings and responses of existing risks, and or if a new risks is a result.

Performance Reports: Performance reports paint a picture of the project's performance with respect to cost, scope, schedule, resources, quality, and risk. Comparing actual performance against baseline plans may unveil risks which may cause problems in the future. Performance reports use bar charts, tables, and histograms, to organize and summarize information such as earned value analysis and project work progress.

3.4.2.3 Risk Assessment .

Software risk assessment is a process of identifying, analyzing, and prioritizing all the possible risks that may occur. As our project is a medium scale project so there is a chance that it will be influenced by multiple risks. So, each project risk needs an assessment strategy so that it does not cause failure or loss of the project. Performing a risk assessment is an important step in being prepared for potential problems that can occur within any software project. During the risk assessment process, a solution or plan of action should be developed for all possible risks.

After analyzing all the risks we have to prioritize the risks according to their severity and impact on the project goals, project scope, timeliness and budget. Table 3.10 shows the probabilities, severity and impact of all the possible risk that may occur during the project.

Table 3.10: Prioritizing All Risks w.r.t Impact

<i>Sr. no</i>	<i>Risk</i>	<i>Category</i>	<i>Probability</i>	Severity	Impact Value	RMMM
1	Failed to meet the requirements.	Project Risk	30 Per-cent	Catastrophic	1	Risk Monitoring
2	Non responsive/unsupportable software	Technical Risk	20 Per-cent	Catastrophic	1	Risk Monitoring
3	Change in requirements may occur.	Project Risk	40 Per-cent	Critical	2	Risk Management
4	Mastering Django which is a new technology may require extra time.	Technical Risk	10 Per-cent	Critical	2	Risk Monitoring
5	There is a possibility that high chart & map API may not support Django framework.	Technical Risk	5 Per-cent	Critical	2	Risk Monitoring
6	Both group members have sufficient development skills but lack experience.	Known Risk	30 Per-cent	Critical	2	Risk Mitigation
7	Shortage of financial resources	Business Risk	10 Per-cent	Critical	2	Risk Management
8	Less reuse than planned	Project Risk	40 Per-cent	Critical	2	Risk Management
9	Insufficient training to use latest tools	Known Risk	20 Per-cent	Marginal	3	Risk Mitigation
10	Possible minor budget over-run	Business Risks	20 Per-cent	Negligible	4	Risk Management

Chapter 4

FUNCTIONAL ANALYSIS AND MODELING

4.1 Use Case Modeling

4.1.1 User Stories

1. Sign-In:

- As a user, I want to login into the system using my email address and password.
- As a system administrator, I want to register new users by entering their personal information.

2. Monitoring of Healthcare Programs:

- As a user, I want select the healthcare program for monitoring from the menu bar.
- As a user, I want to navigate between the healthcare programs whenever I want from the menu bar.
- As a user, I want to select a district after entering the healthcare program to monitor the health status.
- As a user, I want the system to display the healthcare data in a way that it is easily understandable.
- As a user, I want the data to be displayed using graphs or charts or both to assist me in the monitoring process.

3. Area Map:

- As a user, I want an interactive map that allows me to see the health status of a UC after selecting a district.

4. Generate Report:

- As a user, I want to download the healthcare reports district wise in a portable format such as .csv or .xlsx.
- As a user, I want the freedom to select the time span of the report to be downloaded such as monthly report, quarterly report, annual report or custom report.

5. Healthcare Trends:

- As a user, I want to see the current disease trends that can help me in decision making.
- As a user, I want those trends to be displayed in the form of a graph.

6. Disease Prediction:

- As a user, I want the system to perform prediction about a disease that can be threat to a district in future.

7. Update Record:

- As a system administrator, I want to be able to update the healthcare data by uploading the raw healthcare files.
- As a user, I want to be able to update the data district wise for each healthcare program.

4.1.2 Individual Actor Use Cases

4.1.2.1 High Level Use Cases

Use Case #	UC01
Use Case	Login
Actors	User, Admin
Type	Primary, Secondary
Description	For the login Admin has to enter his/her Email and password. If both, Email and Password are correct, then the Admin will be directed to the Main Dashboard otherwise it will display an Error Message.
Use Case #	UC02
Use Case	Register New User
Actors	Admin
Type	Secondary
Description	Admin will have Login into the system using his Email and Password. After that the admin go to Signup form and enter the details of new user along with his/her E-mail and Password. After entering all the information the admin will click on “Register” button.
Use Case #	UC03
Use Case	Update Record
Actors	Admin
Type	Secondary
Description	The Admin can also update the record in the database of the dashboard by uploading the raw healthcare files. The Admin will upload the data files district wise for each of the healthcare program.
Use Case #	UC04
Use Case	Monitor Healthcare Program
Actors	User
Type	Primary
Description	After successful login, the user will be directed to the Main Dashboard. Now the user will select the healthcare program for monitoring, from the menu bar. The user can navigate between the healthcare programs whenever he/she want from the menu bar.

Use Case #	UC05
Use Case	Select District
Actors	User
Type	Primary
Description	After selecting the healthcare program, the user will select the district from the drop down menu to monitor the health status of that district. The system will display the healthcare data in a way that it is easily understandable using graphs or charts or both to assist the user in the monitoring process.

Use Case #	UC06
Use Case	Monitor Health Status
Actors	User
Type	Primary
Description	After selecting the district, the user will be able monitor the health status of that district. The system will display the healthcare data using graphs, charts and density map. The user will interact with the map to see the health status of a UC after selecting a district.

Use Case #	UC07
Use Case	Generate Healthcare Report
Actors	User
Type	Primary
Description	The user can download the healthcare reports, district wise in formats such as .csv or .xlsx. The user will be able to select the time span of the report such as monthly report, quarterly report, annual report or custom report. After selecting the time span the user can download the report by clicking on “Download” button.

Use Case #	UC08
Use Case	Analyze Healthcare Trends
Actors	User
Type	Primary
Description	The user will be able to see and analyze the current disease trends that will help the user in decision making. Those trends will be displayed in the form of a graph using the existed healthcare data.
Use Case #	UC09
Use Case	Disease Forecasting
Actors	User
Type	Primary
Description	The user will be able to see the forecasting results after the system has performed disease forecasting on the basis of existing healthcare data. The system will warn the user about a disease that can be threat to a district in future, which will aid in decision making.

4.1.2.2 Expanded Use Cases**UC no. 1: Register New User**

Actor Actions	System Responsibility
1. User enters his username and password.	2. The system verifies the username and password.
	3. Username and password is correct.
	4. The systems directs the user to his home screen.
5. The user clicks on “Register New User”.	6. The system generates registration form.
7. The user enter the details of the new user and clicks on “Save”.	8. The new user is successfully registered.

UC No.1 Extensions:**3a. Username or password is incorrect.**

1. The user responds to the error.
2. User again enters his username and password.
3. Username and password is correct.

7a. The user forgets to enter some information e.g. Last Name.

1. The system generates an error message “Incomplete Information”.
2. The user responds to the error.
3. User enters the missing information.
4. Registration successful.

UC no. 2: Login

Actor Actions	System Responsibility
1. User enters his username and password.	2. The system verifies the username and password.
	3. Username and password is correct.
	4. The systems directs the user to main dashboard screen.

UC No.2 Extensions:**3a. Username or password is incorrect**

1. The system generates an error message “Incorrect Details”.
2. The user responds to the error.
3. User again enters his username and password.
4. Username and password is correct.

UC no. 3: Monitor Healthcare Program

Actor Actions	System Responsibility
1. User selects the healthcare program from the menu bar.	2. The systems directs the user to the selected healthcare program.
	3. The system displays the data related to that program in the form of graphs and charts.
4. User selects a time span e.g. 2017 to 2020 and clicks on “Submit”.	5. The systems display the data of the years selected by the user.

UC No.3 Extensions:**4a. User does not selects a time and clicks on “Submit”.**

1. The system generates an error message “Invalid Entry”.
2. The user responds to the error.
3. User then selects a time span.
4. The data is displayed by the system.

UC no. 4: Select District

Actor Actions	System Responsibility
1. User selects the district from the drop down menu to monitor the health status of a specific district.	2. The systems directs the user to the selected district.
	3. The system displays the data related to that district in the form of graphs, charts and an interactive map.
4. User selects a time span e.g. 2017 to 2020 and clicks on “Submit”.	5. The systems display the data of the years selected by the user.

UC No.4 Extensions:**4a. User does not select a time and clicks on “Submit”.**

1. The system generates an error message “Invalid Entry”.
2. The user responds to the error.
3. User then selects a time span.
4. The data is displayed by the system.

UC no. 5: Monitor Health Status:

Actor Actions	System Responsibility
1. The user selects the district by clicking on the interactive area map.	2. The system displays the data related to that district categories in the form of Union Councils.
Similarly, the user selects a union council by clicking on the interactive area map.	3. The system displays the data related to that union council in the form of tables, graphs and charts.

UC no. 6: Generate Healthcare Report:

Actor Actions	System Responsibility
1. User selects the healthcare program e.g. TB, from the drop down menu.	
2. User selects the district e.g. Jamshoro.	
3. User selects a time span e.g. 2017 to 2020 and clicks on “Generate”.	4. The system generates the report.
5. The user clicks on “Download Report”.	6. The report is downloaded successfully.

UC No.6 Extensions:**3a. User does not select a time and clicks on “Submit”.**

1. The system generates an error message “Invalid Entry”.
2. The user responds to the error.
3. User then selects a time span.
4. The report is generated successfully.

UC no. 7: Analyze Healthcare Trends

Actor Actions	System Responsibility
1. User clicks on “Trends Analysis” from the menu bar of the dashboard.	2. The systems directs the user to the “Trends Analysis” page.
3. User selects the district e.g. Jamshoro.	
4. User selects a time span e.g. 2015 to 2020 and clicks on “Generate”.	5. The system displays the current healthcare trends in the form of a graph.

UC no. 8: Disease Forecasting

Actor Actions	System Responsibility
1. User clicks on “Disease Forecasting” from the menu bar of the dashboard.	2. The systems directs the user to the “Disease Forecasting” page.
3. User selects the district e.g. Jamshoro.	4. The system generates a report that will warn the user about a disease that can be threat to a district in future.

UC no. 9: Update Record

Actor Actions	System Responsibility
1. User clicks on “Update Record” from the menu bar of the dashboard.	2. The systems directs the user to the “Update Record” page.
3. User selects the healthcare program for which record is being updated e.g. LHWP.	
4. User selects the district for which record is being updated e.g. Benazirabad.	
5. The user selects the file from the system and clicks on “Update Record”.	6. The record is updated successfully in the database of the dashboard.

UC No.9 Extensions:**5a. User does not selects a file and clicks on “Update Record”.**

1. The system generates an error message “No file Selected”.
2. The user responds to the error and selects a file.
3. The record is updated successfully.

5b. User selects an invalid file e.g. a “.txt file”.

1. The system generates an error message “Invalid File”.
2. The user responds to the error and selects a valid file.
3. The record is updated successfully.

4.1.3 Use Case Diagram

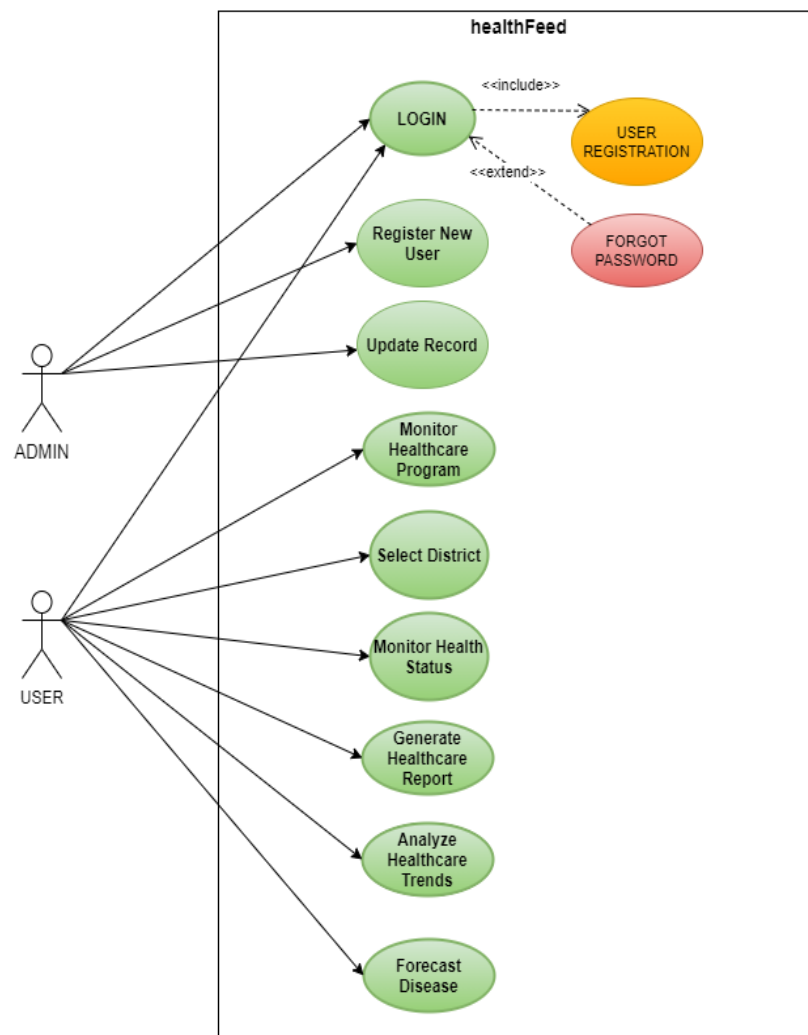


Figure 4.1: Use Case Diagram

4.2 Functional Modeling

4.2.1 Entity Relationship Diagram

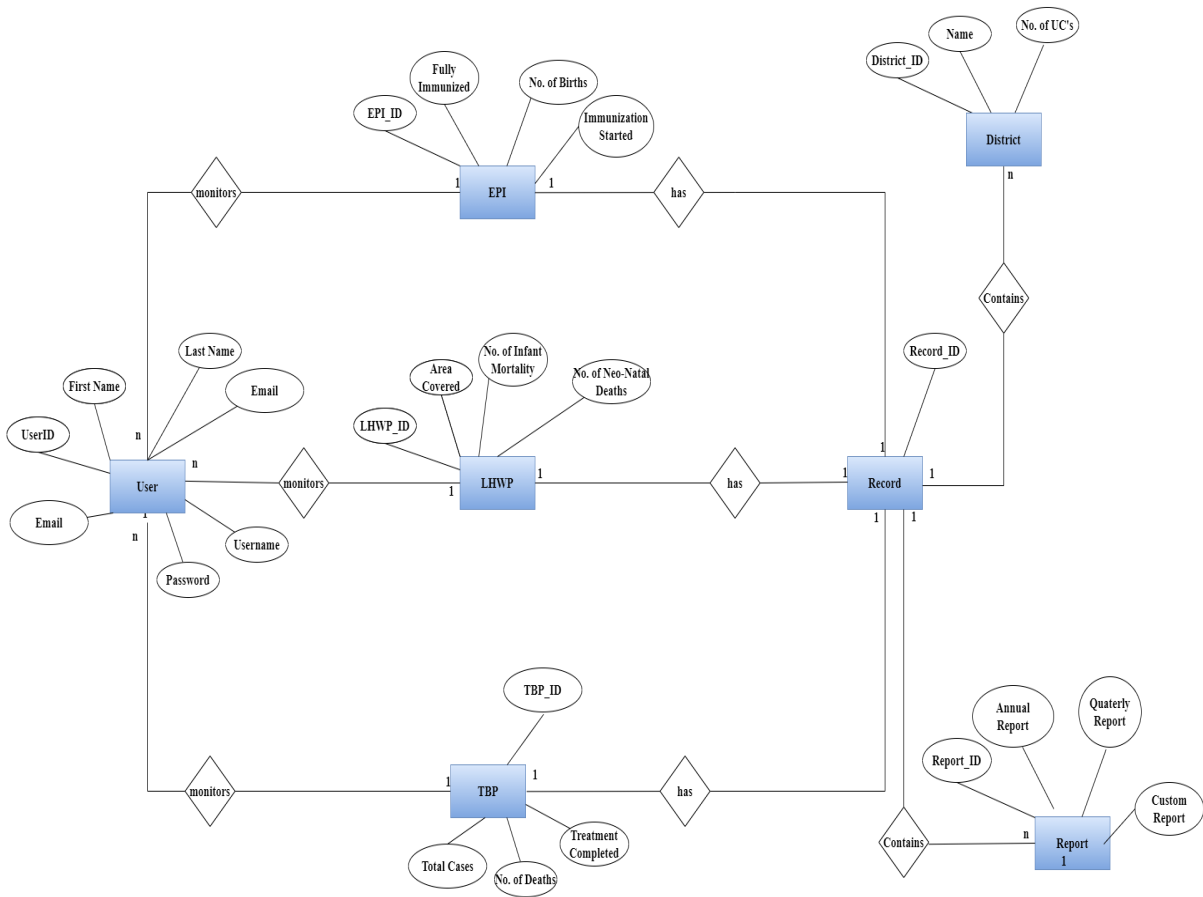


Figure 4.2: Entity Relationship Diagram

4.2.2 Relational Schema

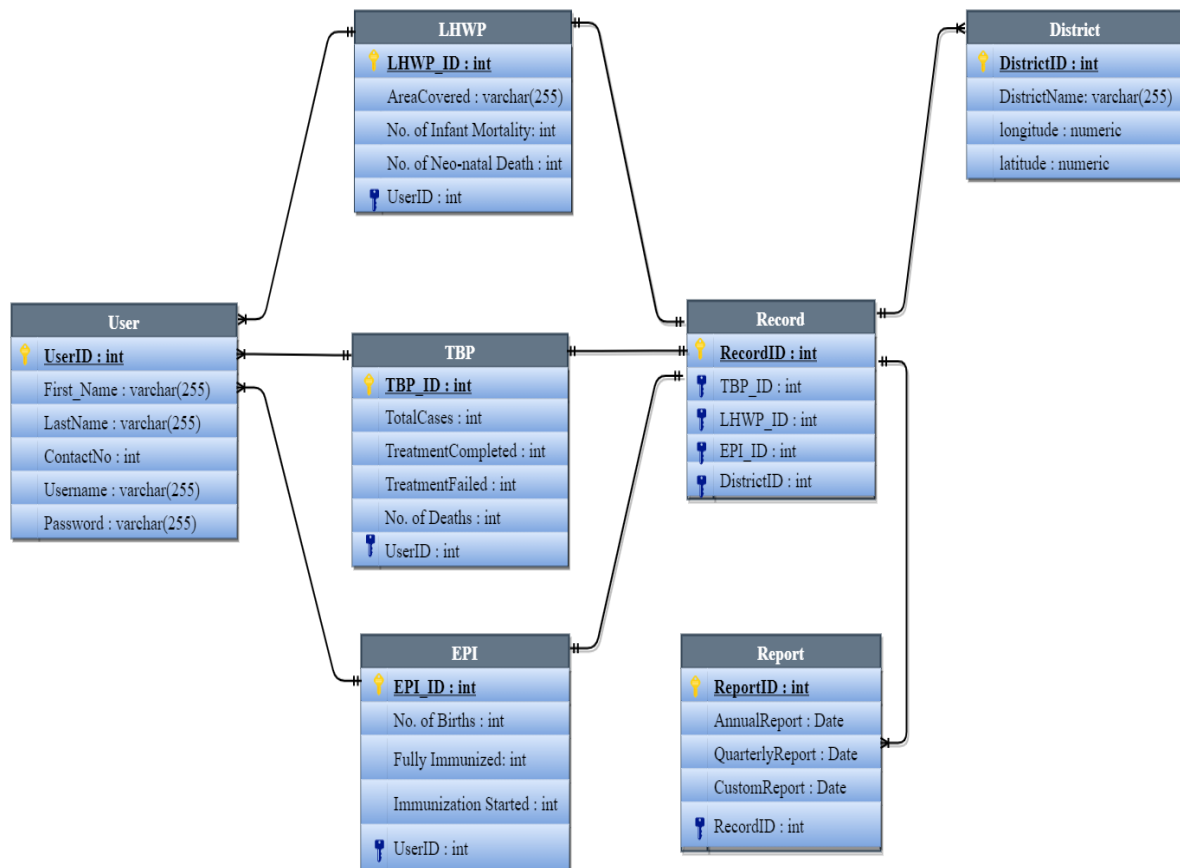


Figure 4.3: Relational Schema

4.2.3 Data Flow Diagram

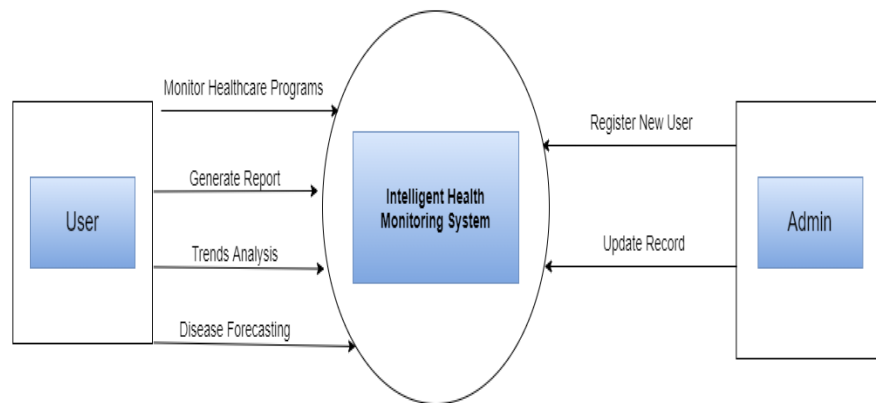


Figure 4.4: DFD Level 0

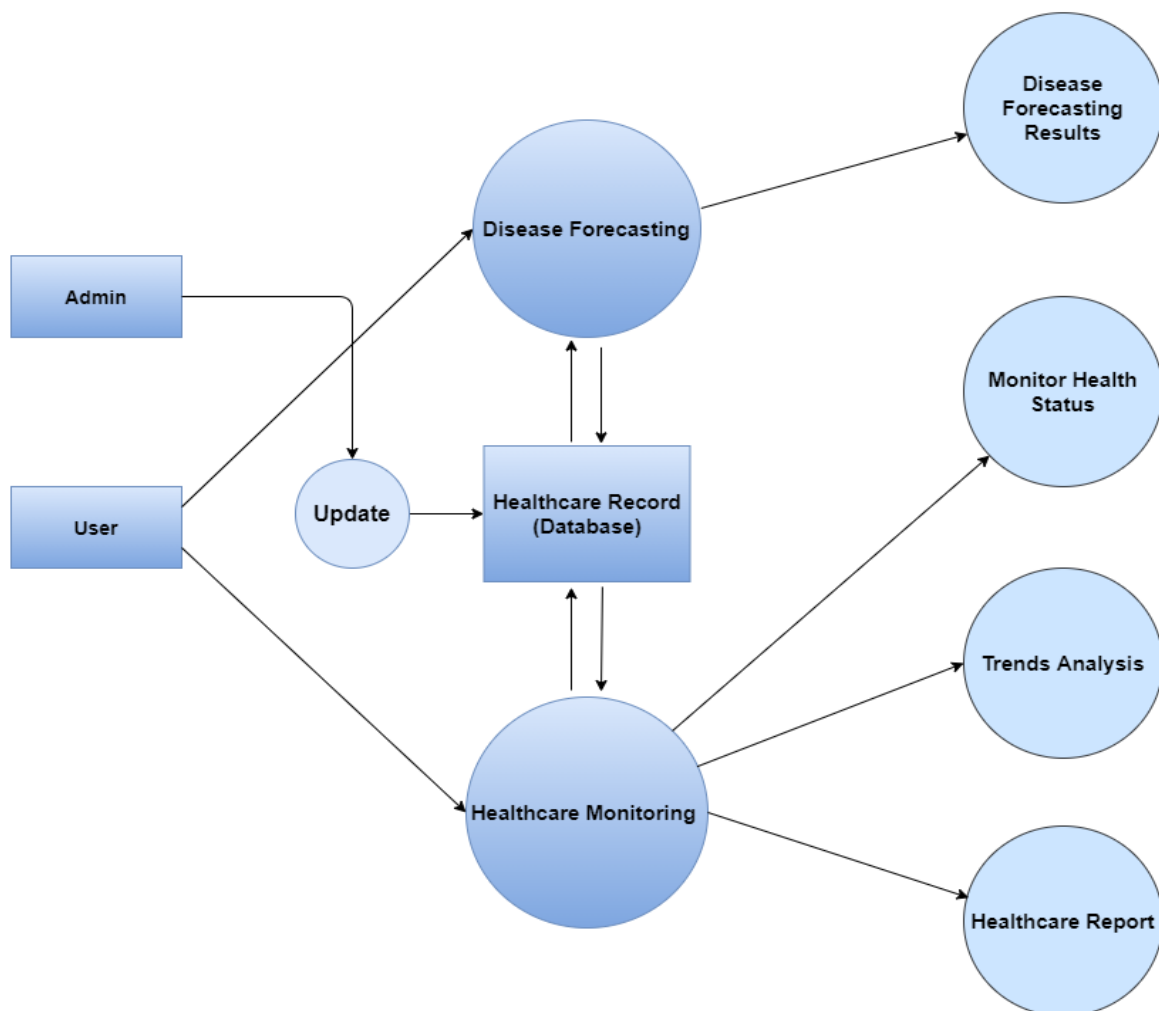


Figure 4.5: DFD Level 1

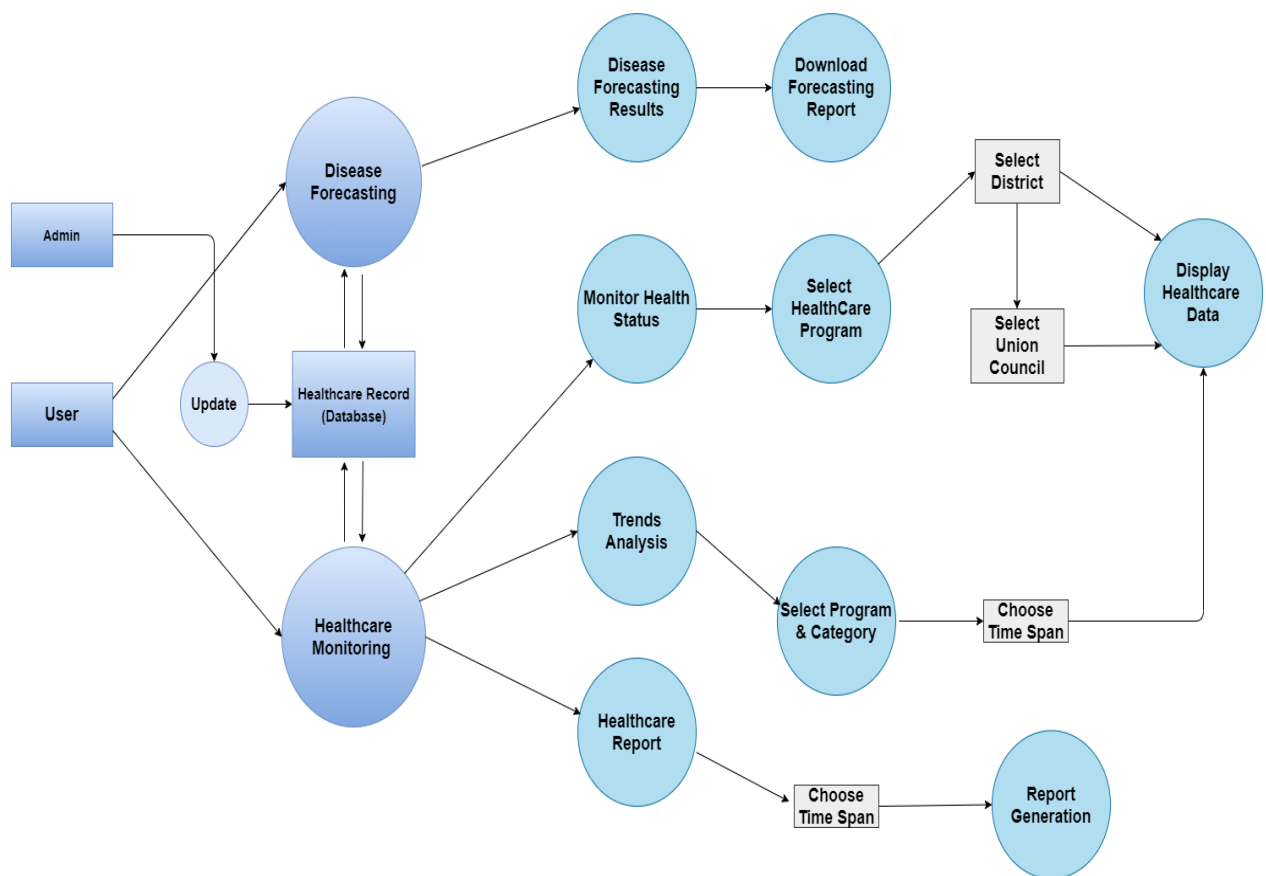


Figure 4.6: DFD Level 2

Chapter 5

SYSTEM DESIGN

5.1 Structure Diagrams

5.1.1 Class Diagram

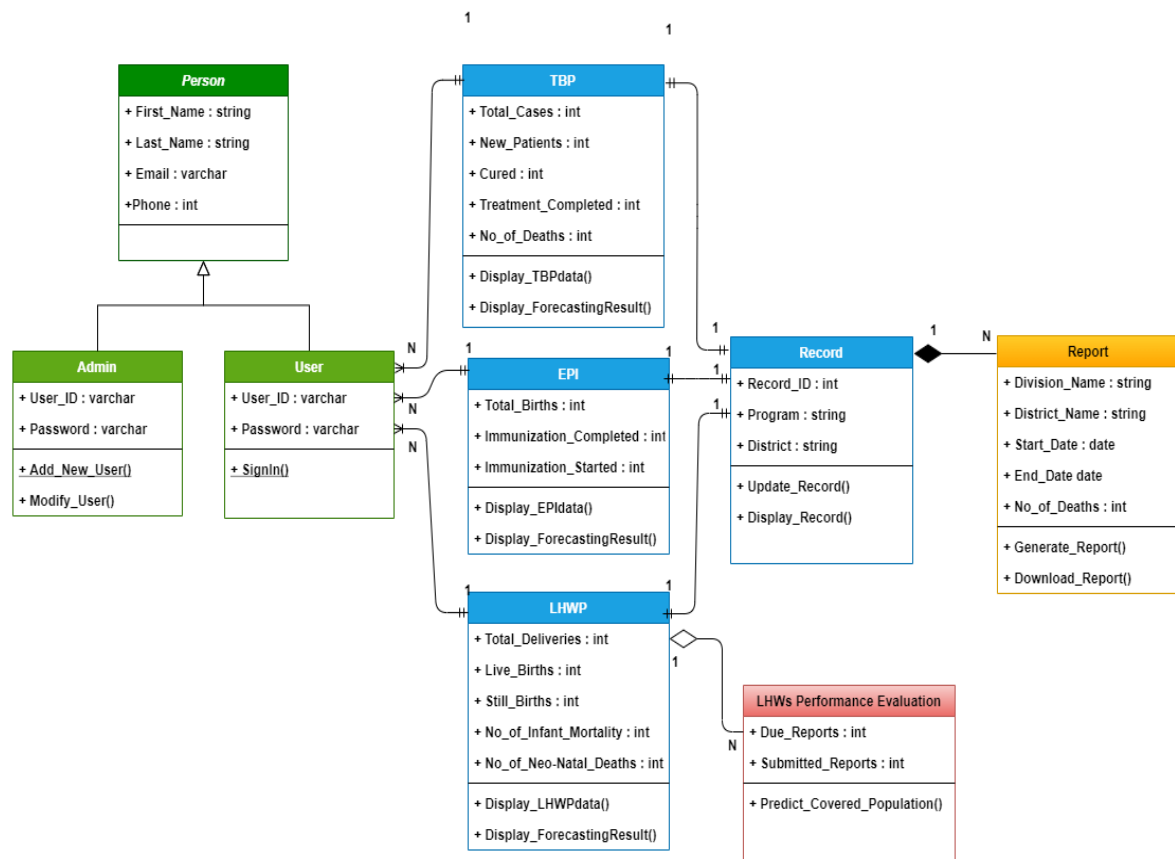


Figure 5.1: Class Diagram

5.1.2 Deployment Diagrams

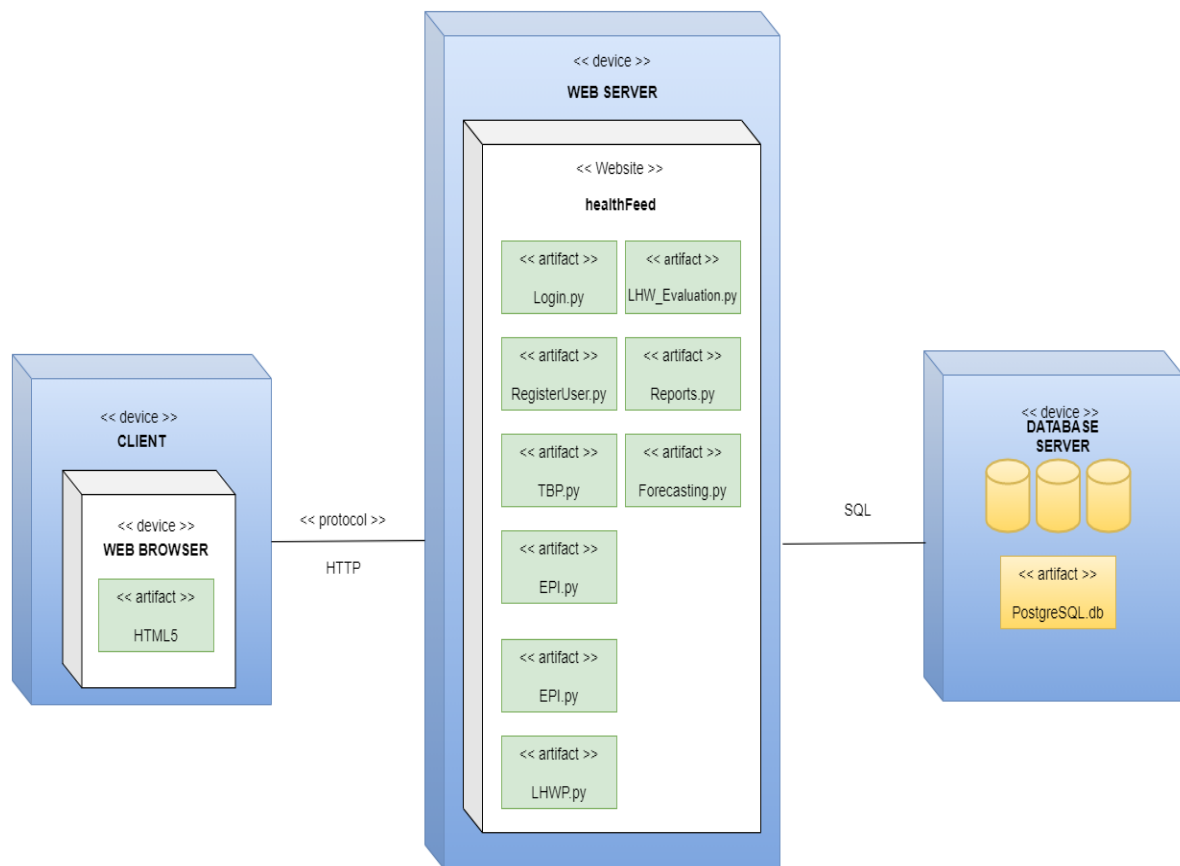


Figure 5.2: Deployment Diagram

5.2 Behavioral Diagrams

5.2.1 Activity Diagrams

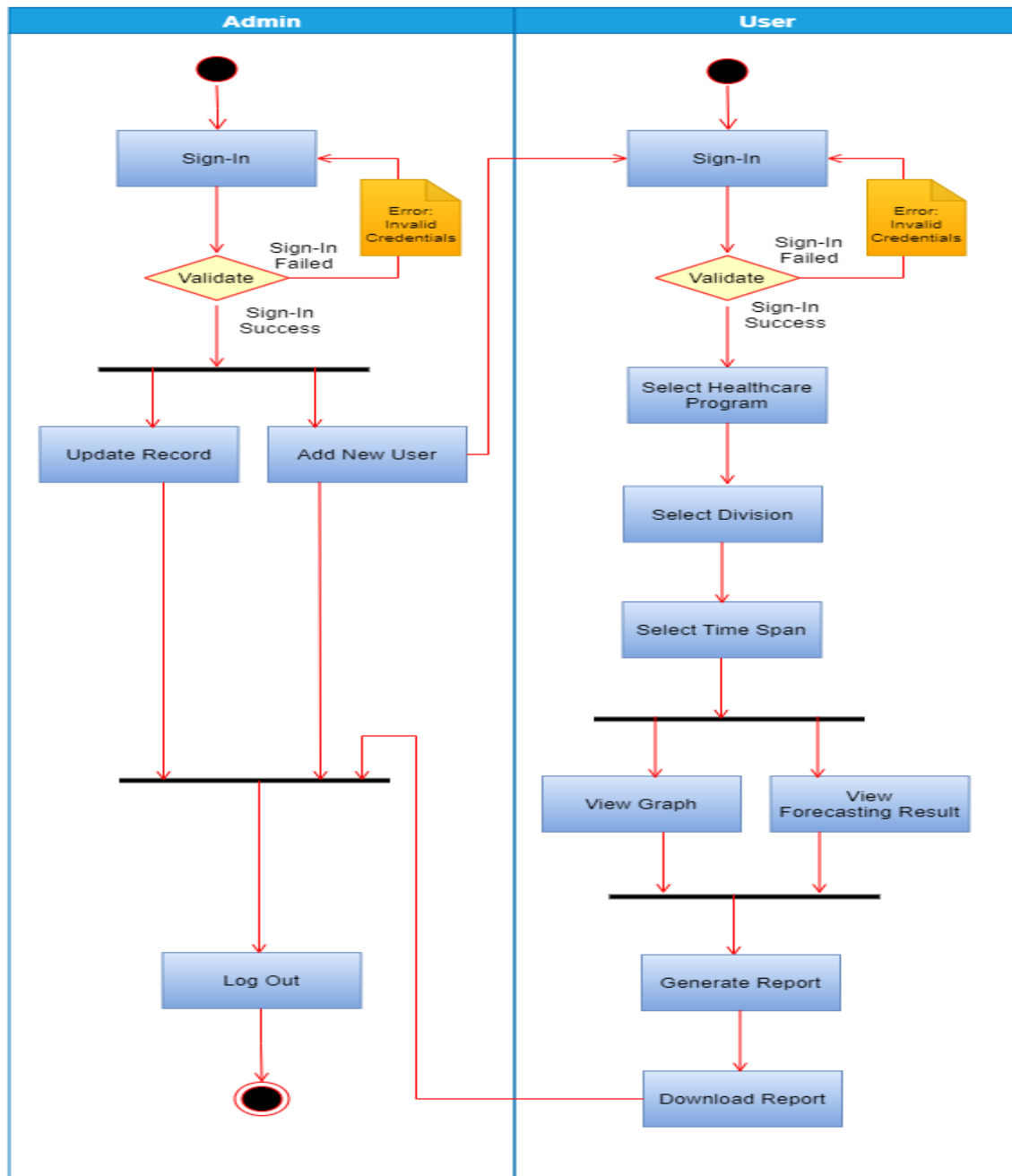


Figure 5.3: Activity Diagram

5.2.2 Communication Diagrams

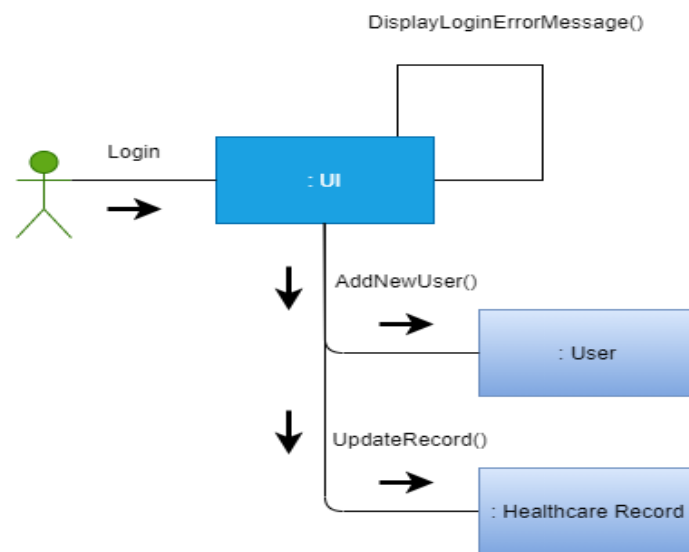


Figure 5.4: Communication Diagram: Admin

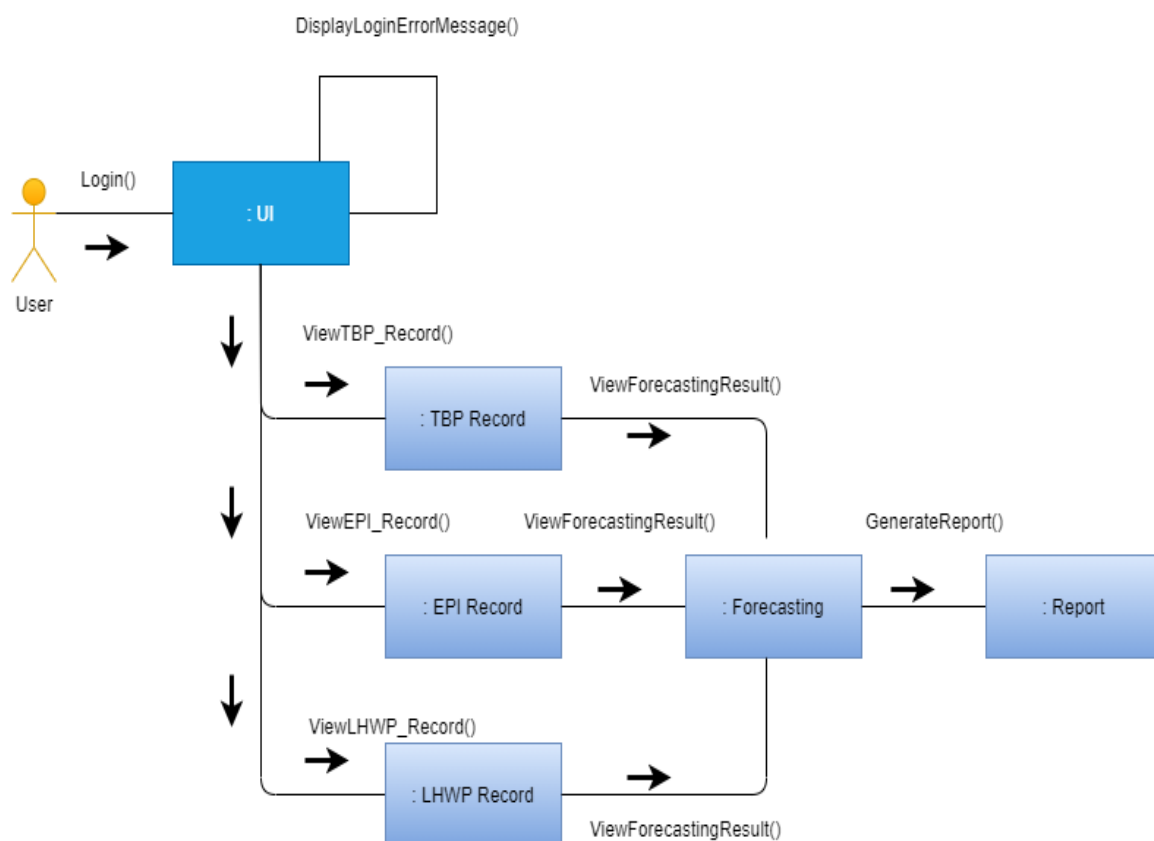


Figure 5.5: Communication Diagram: User

5.2.3 Sequence Diagrams

5.2.3.1 Admin Login

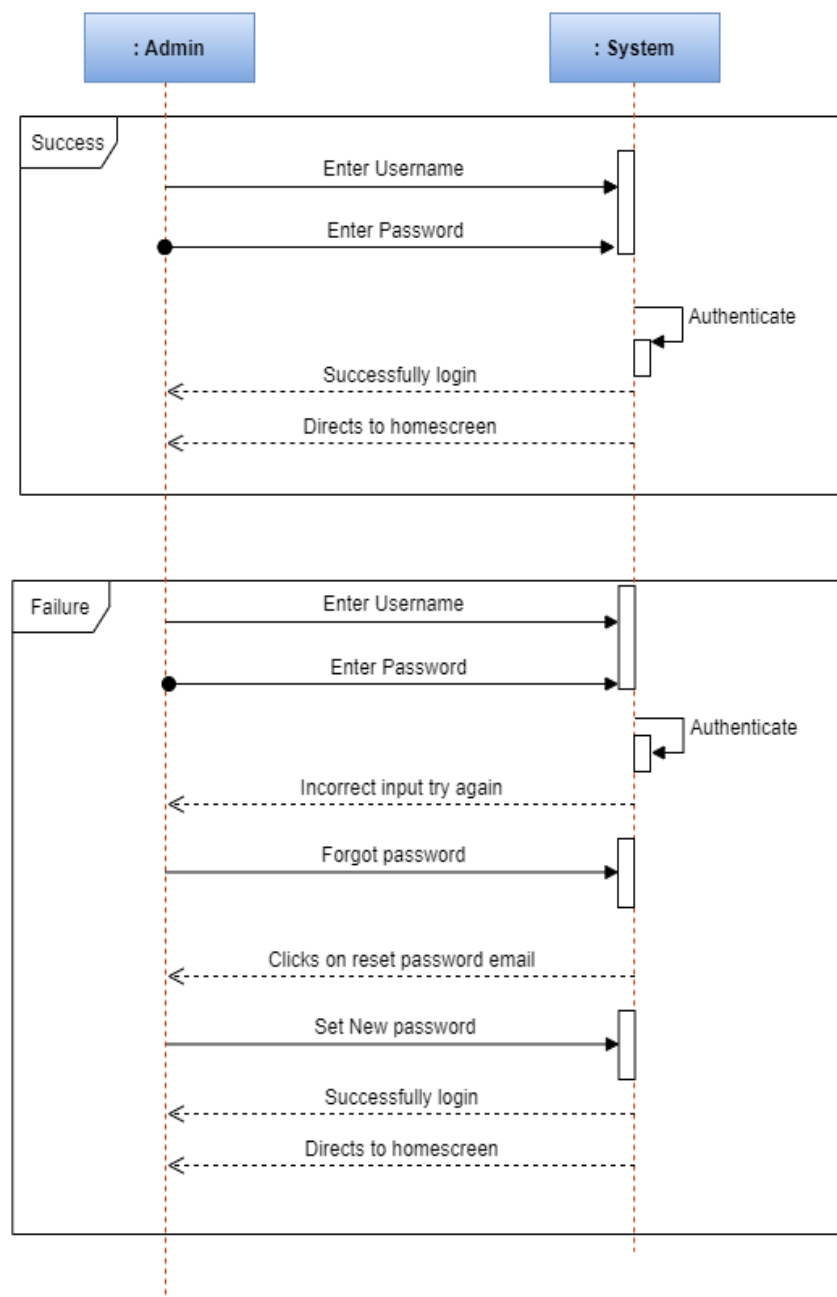


Figure 5.6: Admin: Login

5.2.3.2 Add New User

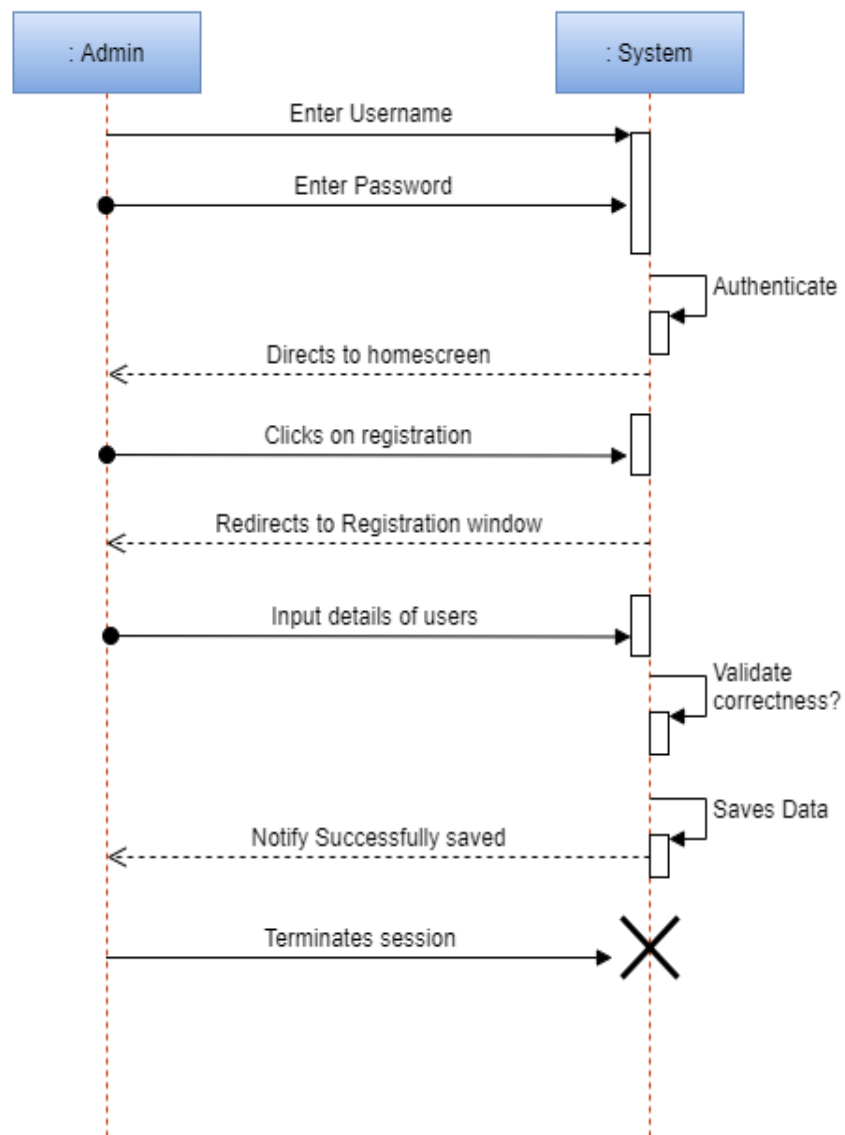


Figure 5.7: Admin: Add New User

5.2.3.3 Update Record

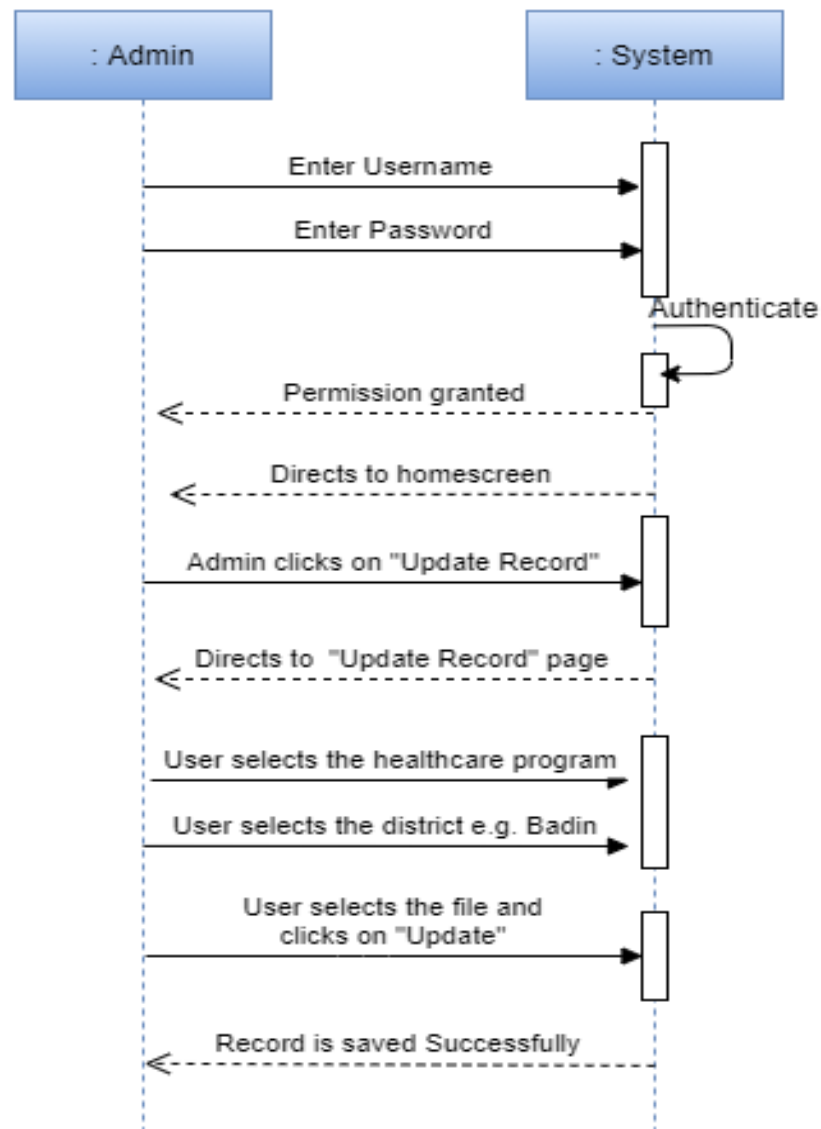


Figure 5.8: Admin: Update Record

5.2.3.4 User Login

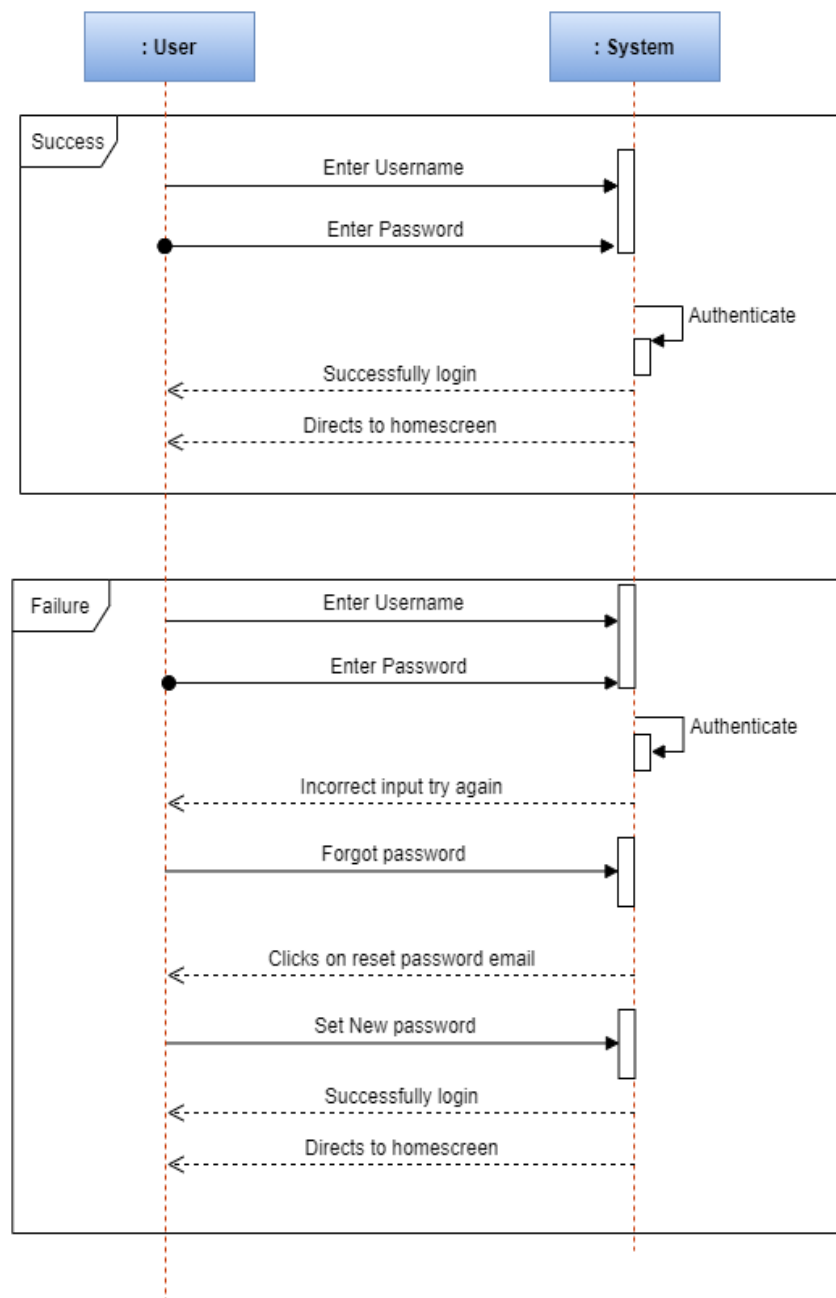


Figure 5.9: User: Login

5.2.3.5 Monitor Healthcare Program

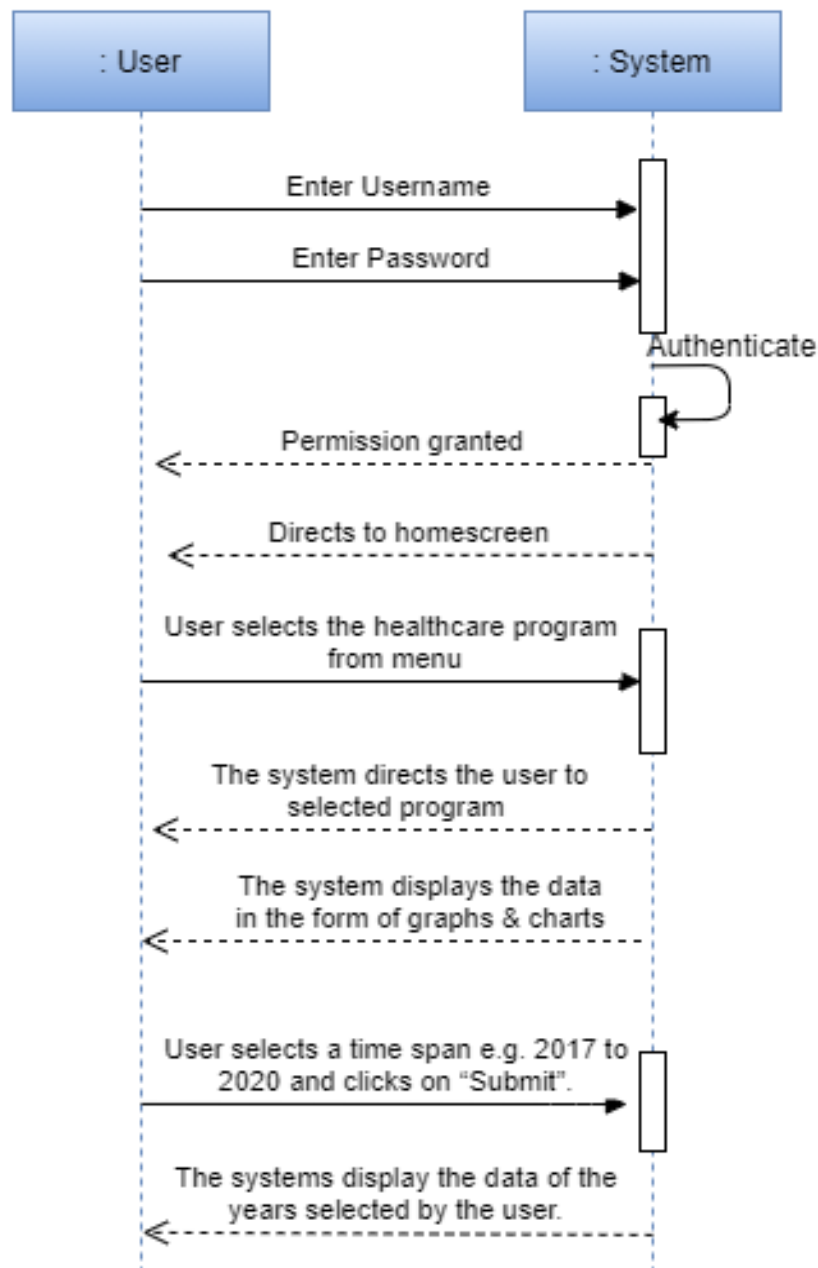


Figure 5.10: User: Monitor Healthcare Program

5.2.3.6 Select District

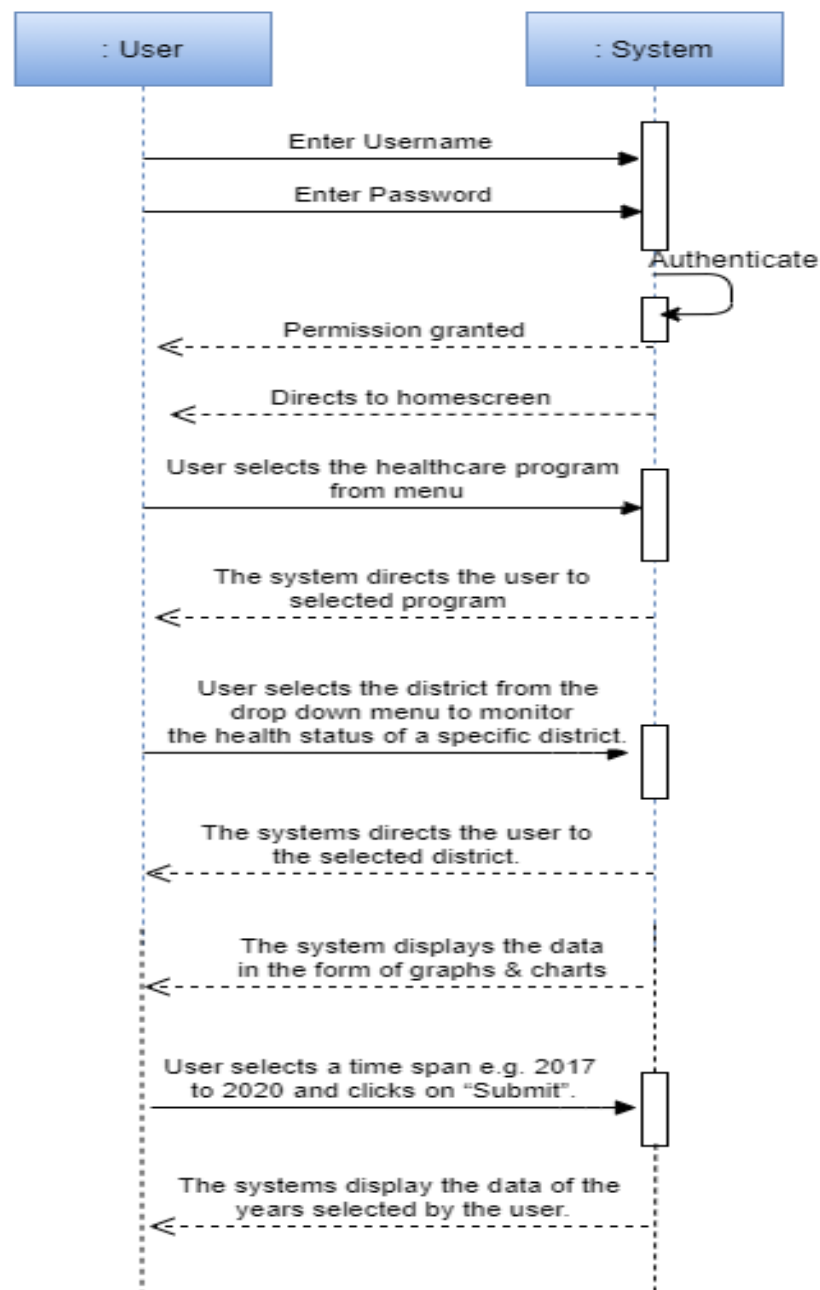


Figure 5.11: User: Select District

5.2.3.7 Monitor Health Status

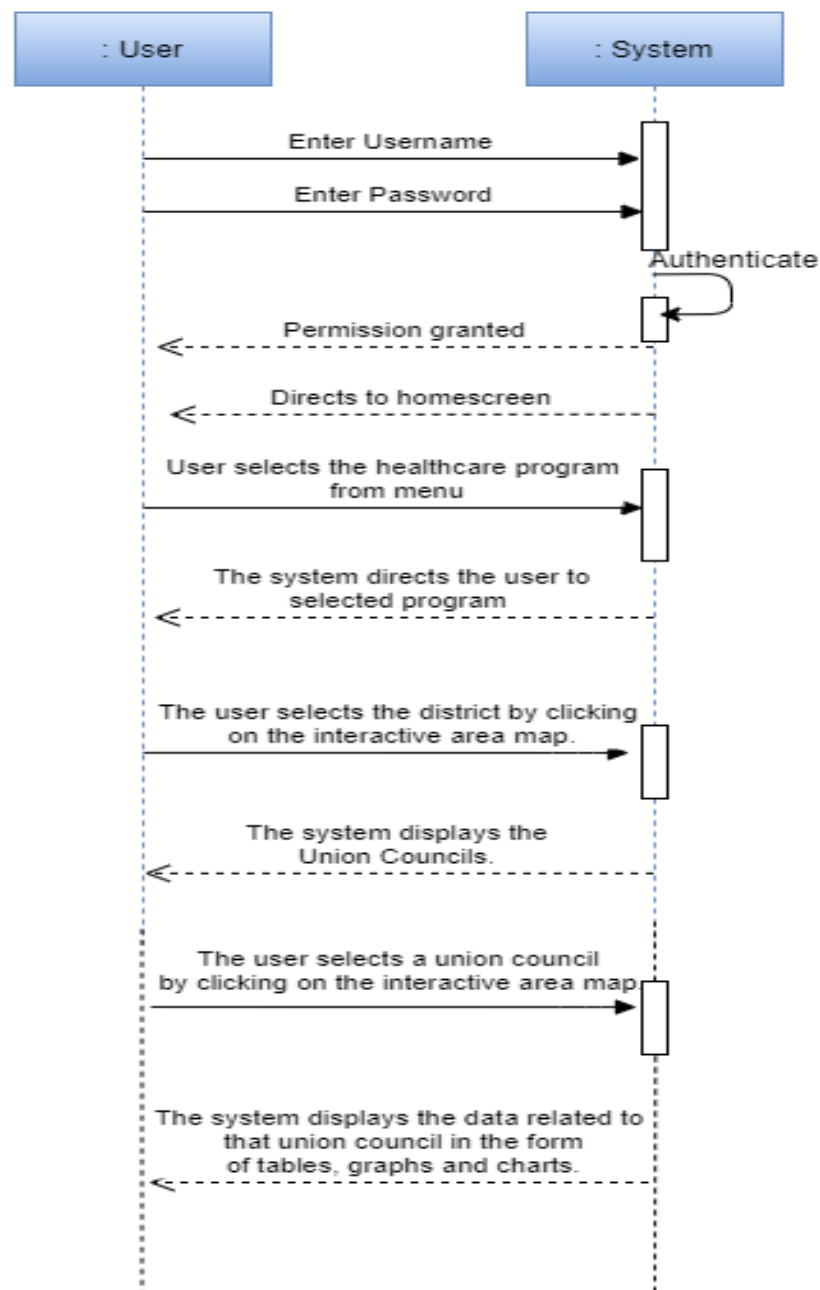


Figure 5.12: User: Monitor Health Status

5.2.3.8 Generate Healthcare Report

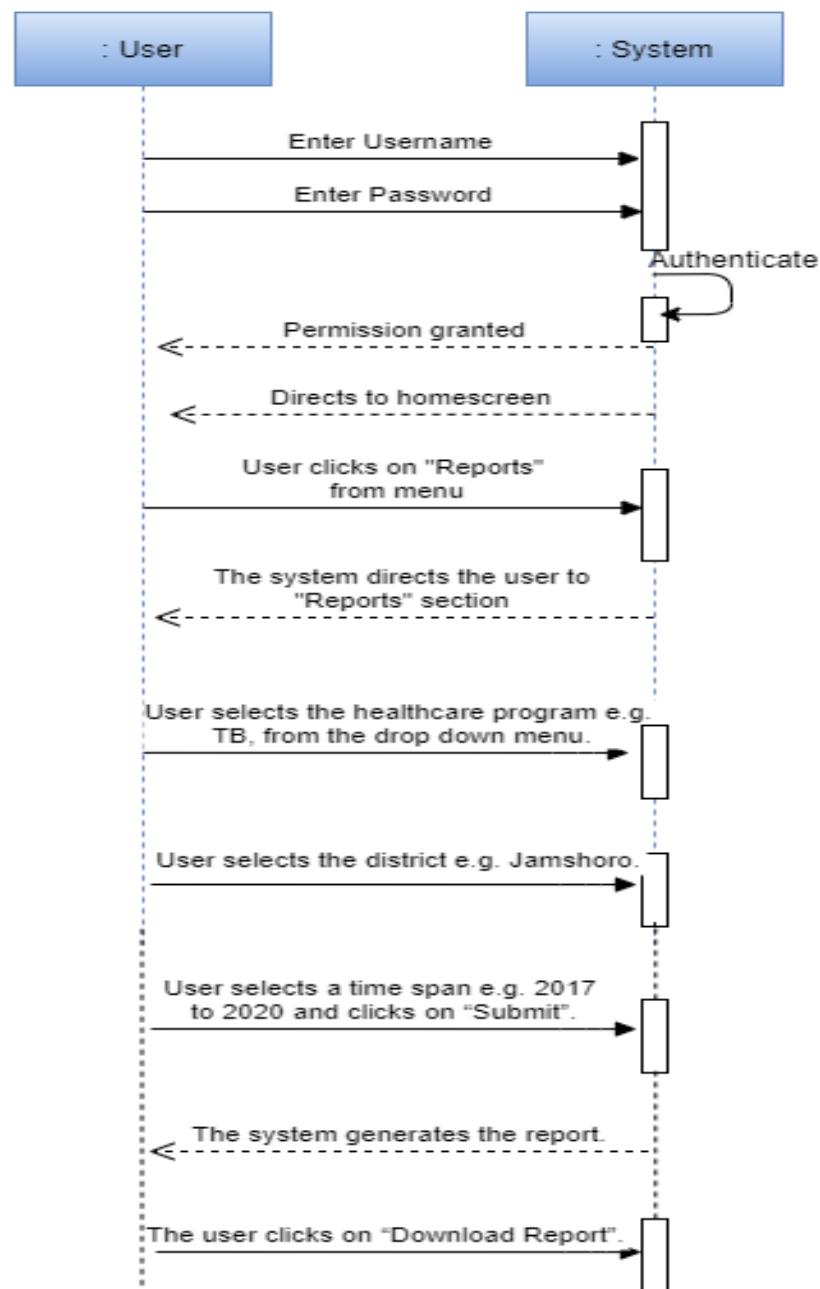


Figure 5.13: User: Generate Healthcare Report

5.2.3.9 Analyze Healthcare Trends

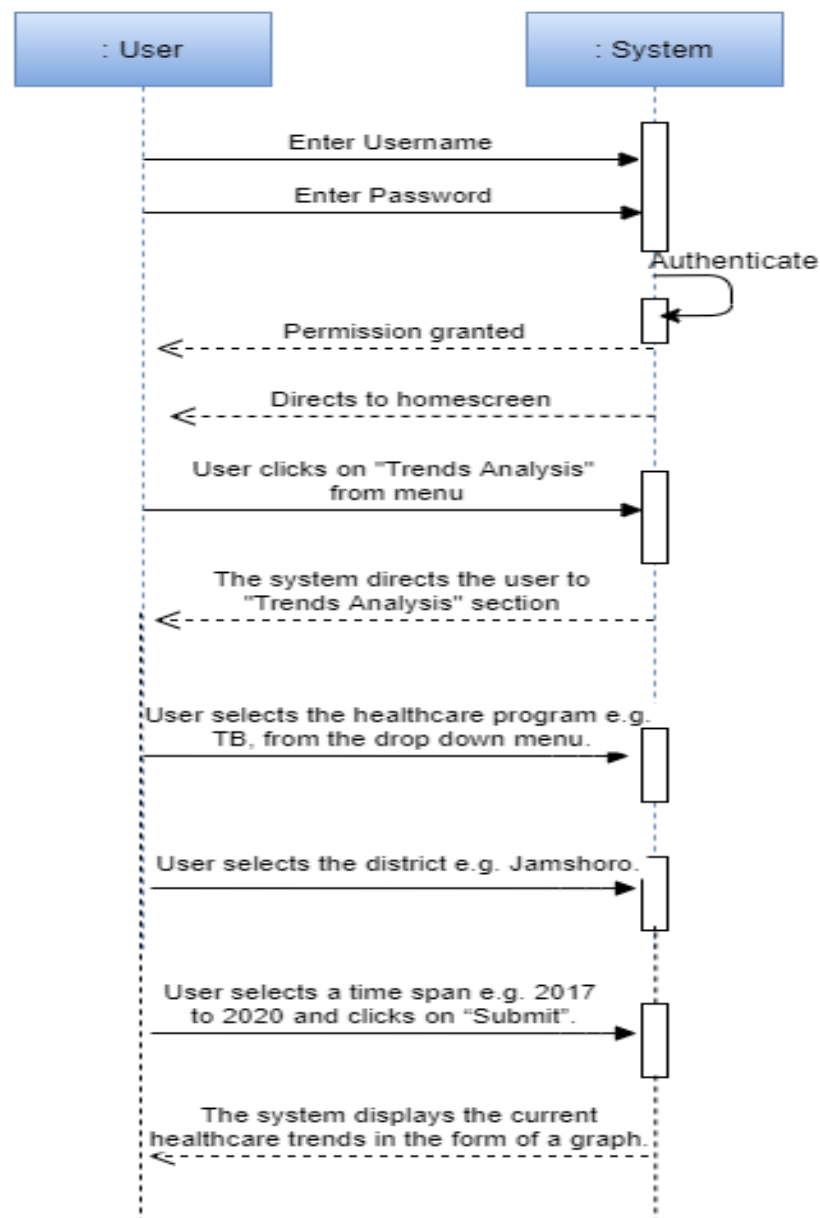


Figure 5.14: User: Analyze Healthcare Trends

5.2.3.10 Disease Forecasting

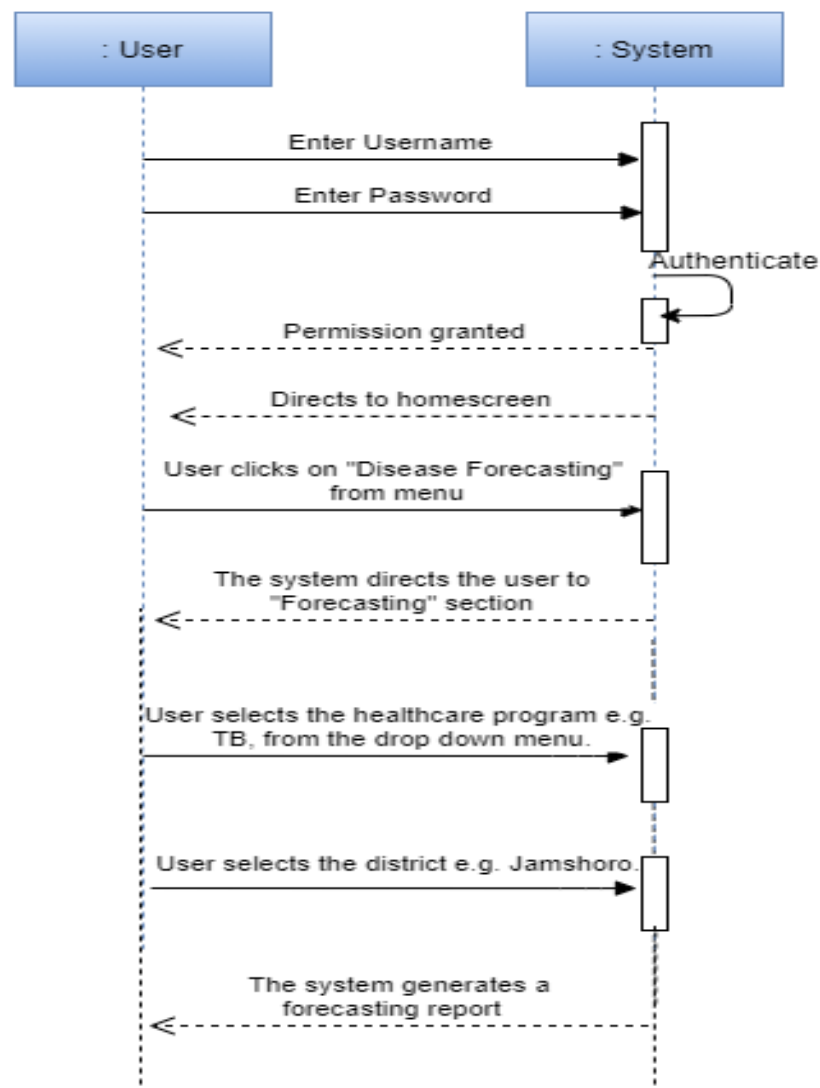


Figure 5.15: User: Disease Forecasting

Chapter 6

SYSTEM INTERFACE AND PHYSICAL DESIGN

6.1 System User Interfaces

6.1.1 Home Page

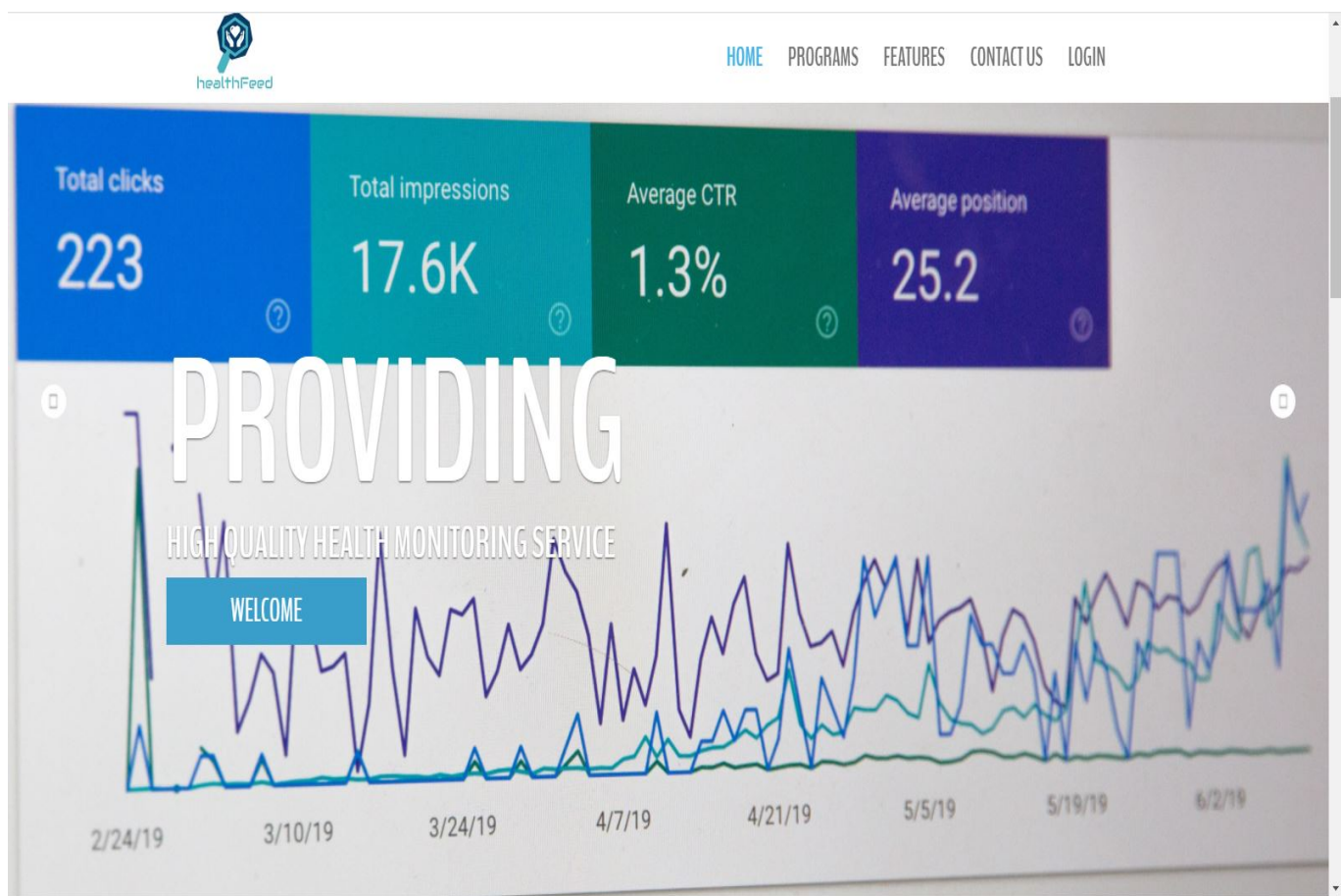


Figure 6.1: Home Page

6.1.2 Home Page

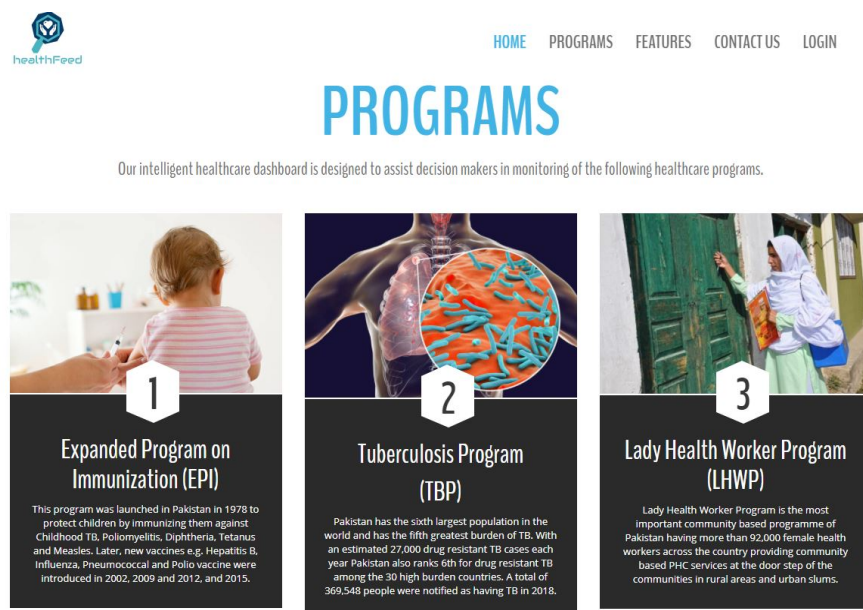


Figure 6.2: Home Page

6.1.3 Sign-In Interface

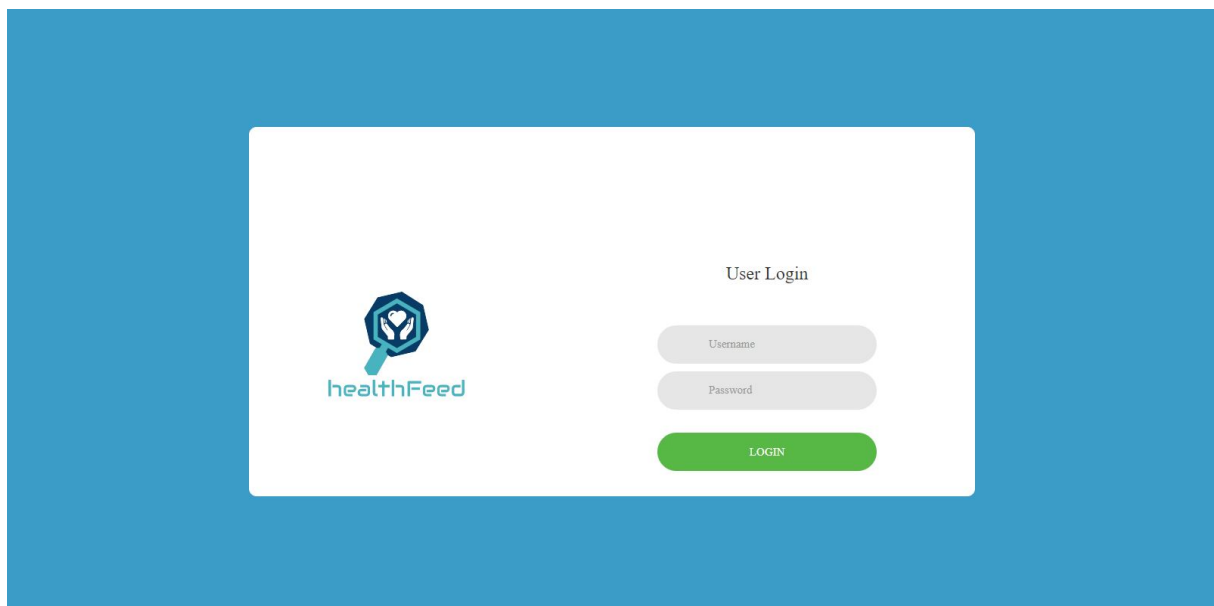


Figure 6.3: Sign-In Interface

6.1.4 TB Section

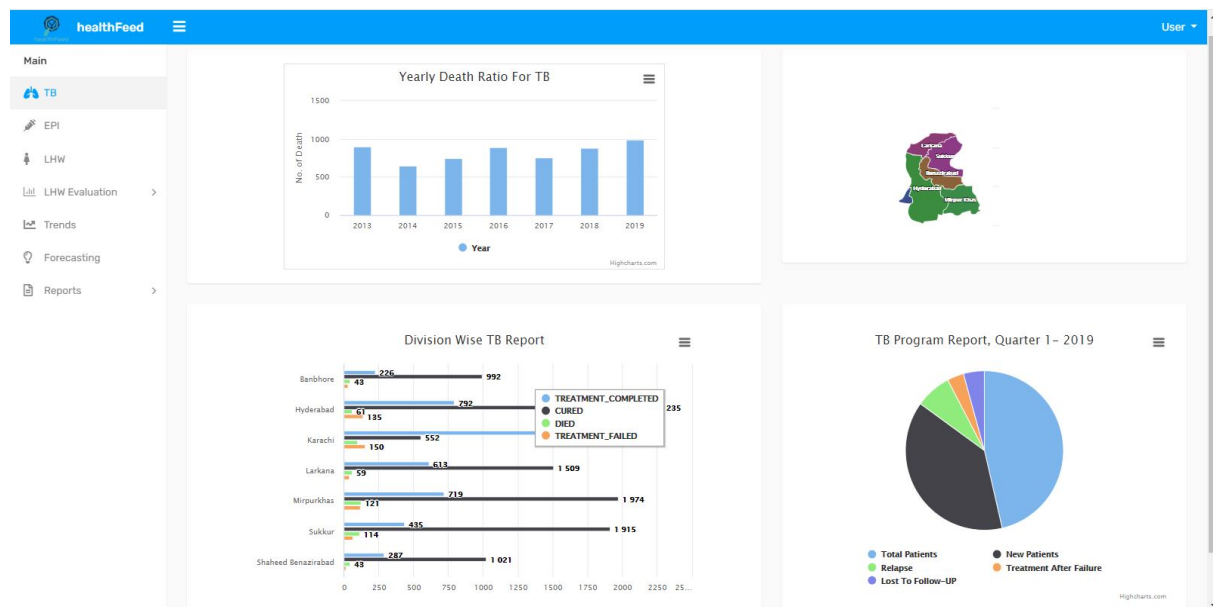


Figure 6.4: TB Section

6.1.5 EPI Section



Figure 6.5: EPI Section

6.1.6 LHW Section

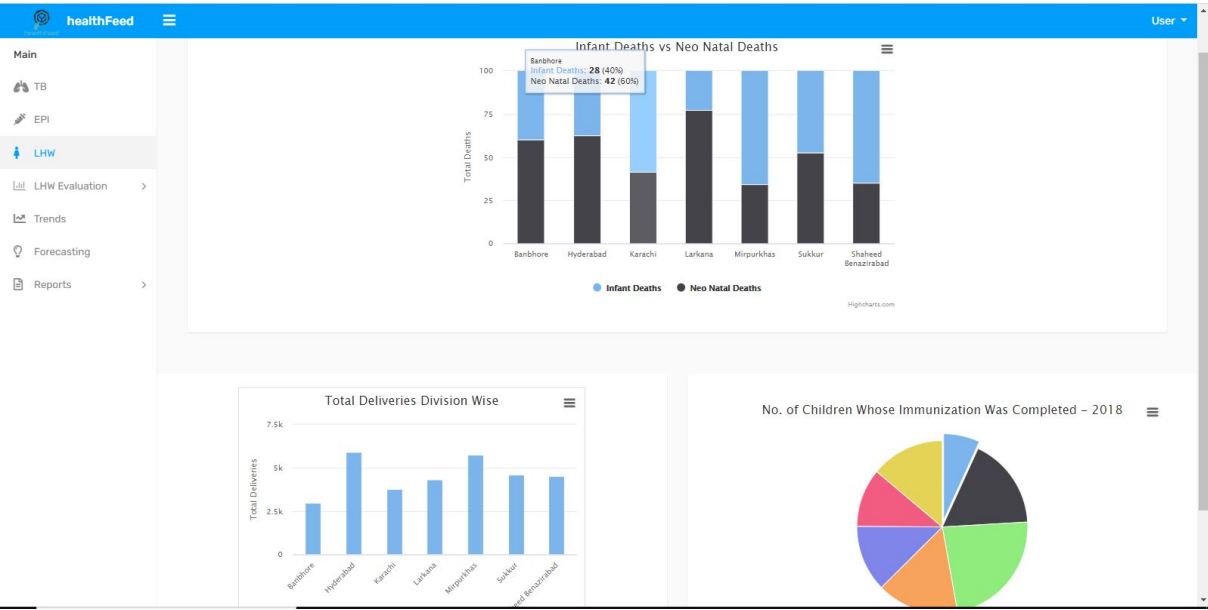


Figure 6.6: LHW Section

6.1.7 LHW Evaluation: Area Wise

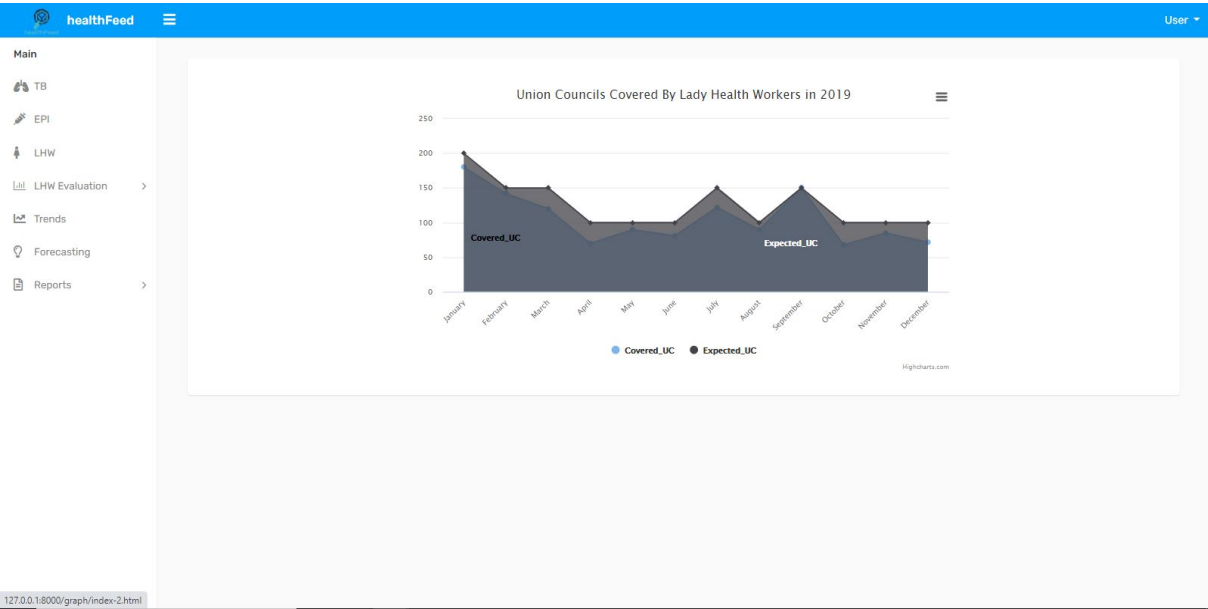


Figure 6.7: LHW Evaluation: Area Wise

6.1.8 LHW Evaluation: Report Wise



Figure 6.8: LHW Evaluation: Report Wise

6.1.9 Trends Analysis

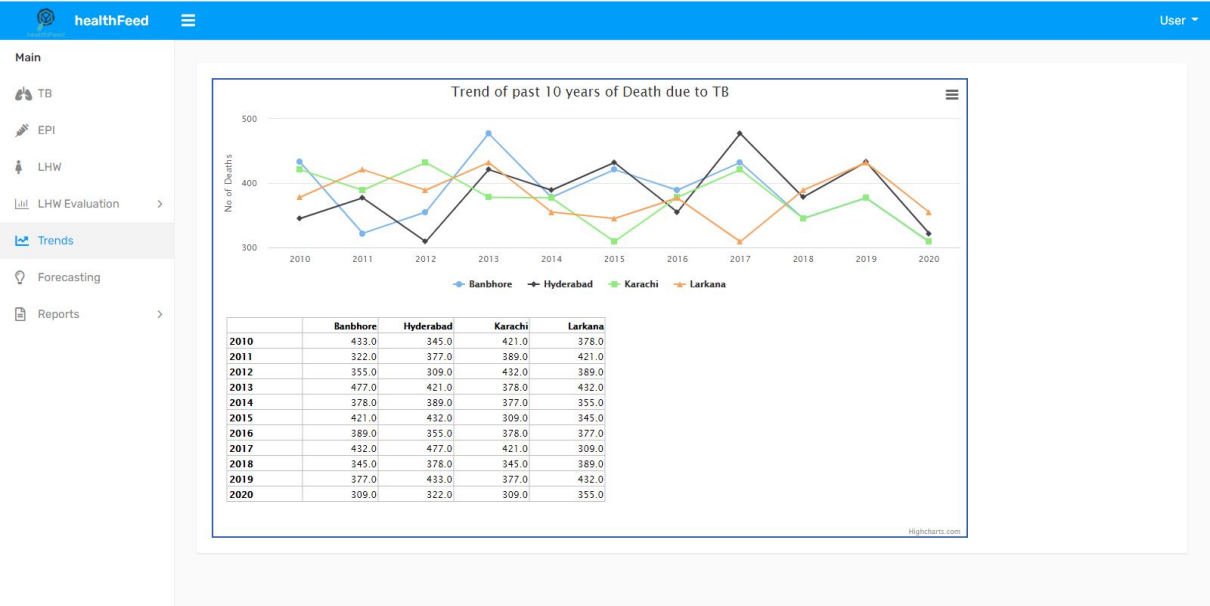


Figure 6.9: Trends Analysis

6.1.10 Sign-Up Interface

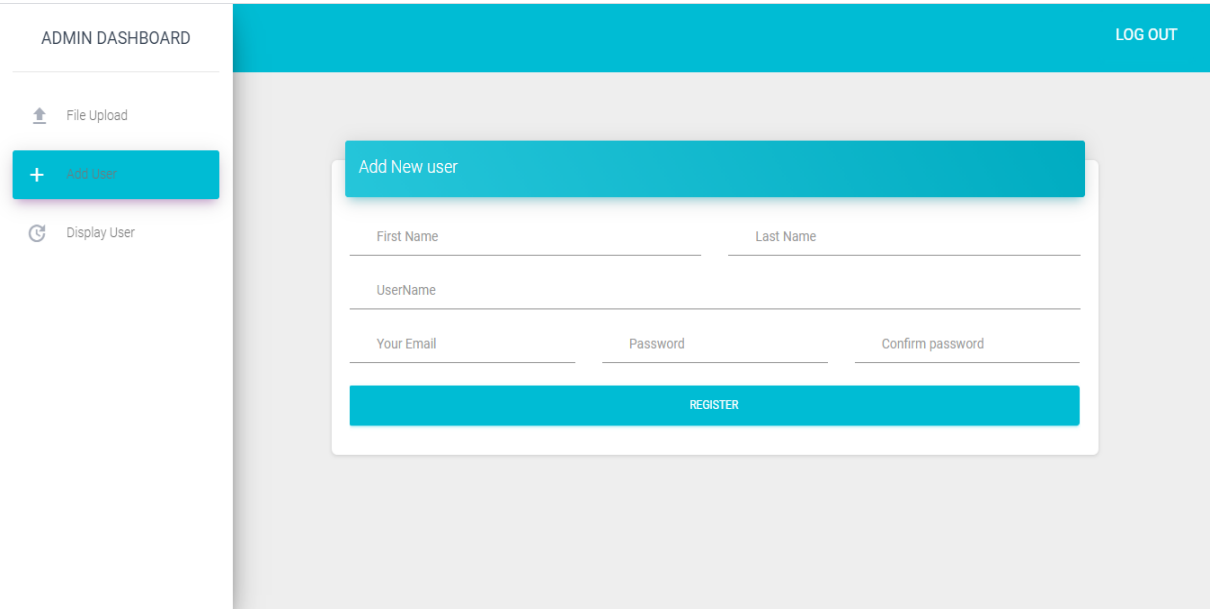


Figure 6.10: Sign-Up Interface

6.1.11 Update Record Interface

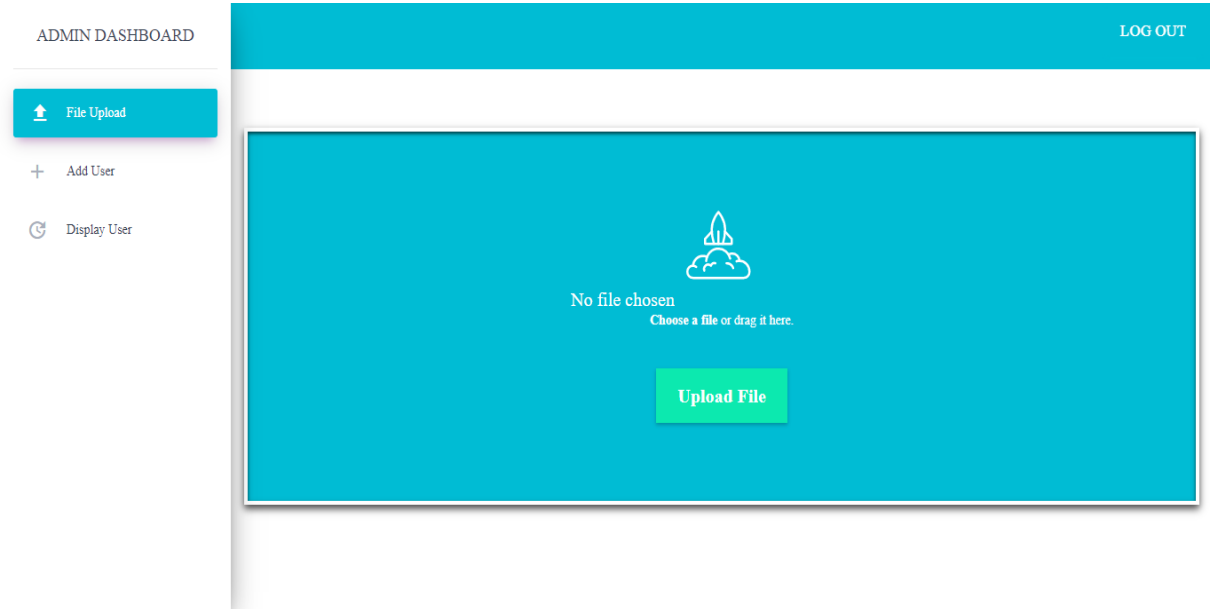


Figure 6.11: Update Record Interface

6.1.12 Admin Table

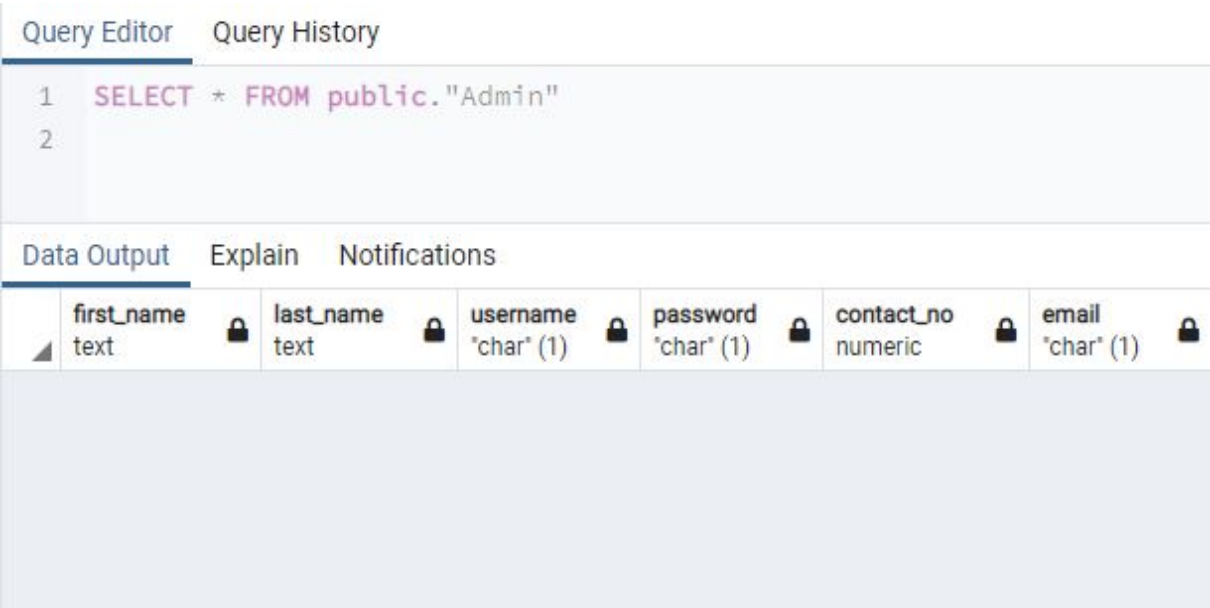


Figure 6.12: Admin Table

6.1.13 User Table

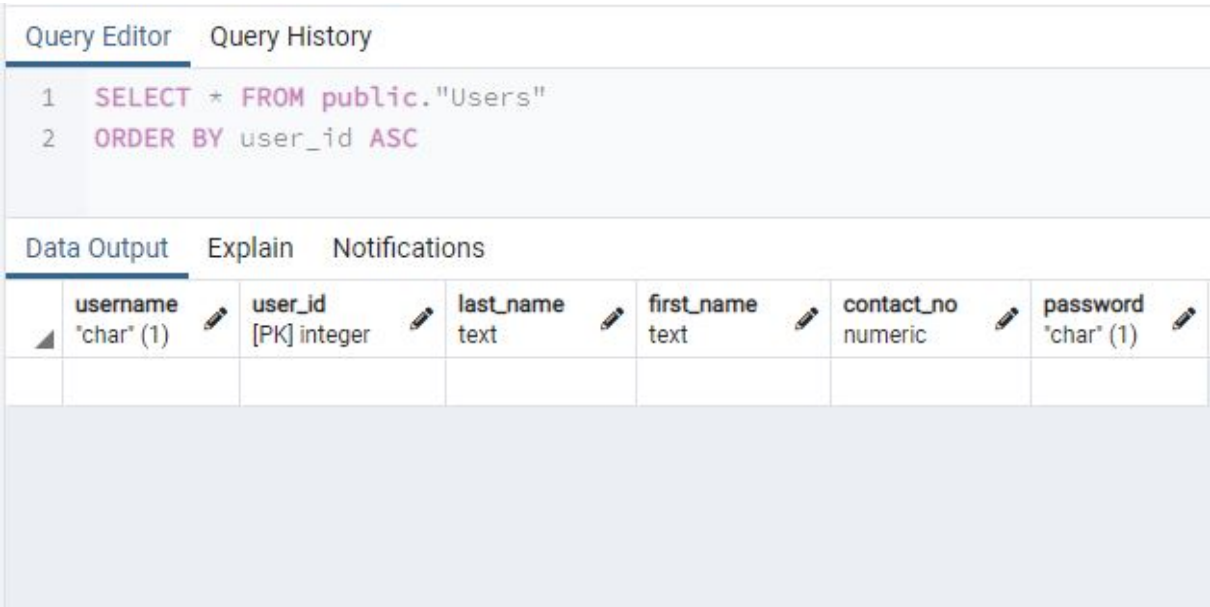


Figure 6.13: User Table

6.1.14 TBP Table

Query Editor

Query History

1

SELECT * FROM public."TBP"

2

ORDER BY tbp_id ASC

Data Output

Explain

Notifications

	<div>tbp_id</div> <div>[PK] integer</div>	<div>total_cases</div> <div>integer</div>	<div>treatment_completed</div> <div>integer</div>	<div>treatment_failed</div> <div>integer</div>	<div>no_deaths</div> <div>integer</div>

Figure 6.14: TBP Table

6.1.15 EPI Table

Query Editor

Query History

1

SELECT * FROM public."EPI"

2

ORDER BY epi_id ASC

Data Output

Explain

Notifications

	epi_id [PK] integer	no_births integer	fully_immunized integer	immunization_started integer

Figure 6.15: EPI Table

6.1.16 LHWP Table

Query EditorQuery History

1SELECT * FROM public."LHWP"

2ORDER BY lhwp_id ASC

Data OutputExplainNotifications

	lhwp_id [PK] integer	area covered numeric	no_infant_mortality integer	no_neo_natal_death integer

Figure 6.16: LHWP Table

6.1.17 Record Table

Query EditorQuery History

1SELECT * FROM public."Record"

2ORDER BY record_id ASC

Data OutputExplainNotifications

	record_id [PK] integer	tbp_tbp_id integer	lhwp_lhwp_id integer	epi_epi_id integer	district_district_id integer

Figure 6.17: Record Table

6.1.18 Report Table

Query Editor

Query History

1

2

SELECT * FROM public."Report"

ORDER BY report_id ASC

Data Output

Explain

Notifications

	report_id [PK] integer		record_record_id integer		district_district_id integer	

Figure 6.18: Report Table

Chapter 7

TEST PLAN

7.1 Objective of the Testing Phase

In the software testing phase our aim is evaluate the quality of our software product and also to improve it by identifying defects. The ultimate goal of software testing is to recognize and troubleshoot all the present issues and bugs as well as control the quality of our resulted software product. Other than that, there are another two fundamental purposes for which testing is important, first one involves verifying procurement specifications and second one is managing risk. Testing phase will help us to verify and ensure that our software product meets all the functional, design, implementation, and performance requirements identified in the specification document.

Following are the most essential software testing objectives of the testing phase:

1. To verify the fulfillment of all specified requirements:

Our first objective of this phase is to ensure that the final software product fulfills the customer's needs. We must test the product and ensure that all the requirements specified by the customer have been implemented. We have developed all the possible test cases to ensure verification of all functionality demanded by the customer or stakeholder. Requirement Trace-ability Matrix (RTM) is another method to ensure that test cases have got the right requirement coverage.

2. To detect and remove bugs in software product:

As a software tester our objective is to detect and remove defects in the system at the earliest stage of development. The detection and removal of bug is superior objective to others because, bug prevention helps to increase the customer satisfaction on the product, minimize the product time to market, and most importantly reduce the cost of software quality maintenance.

3. To control Software quality and reliability:

Another key objective of software testing is to keeping control of software quality and reliability. To control software quality we have to keep the bugs as minimum as possible and also make sure that our software is compatible with the latest hardware and software including all types web browsers and operating systems etc.

4. To provide sufficient information to stakeholders:

Another objective of the testing phase is to provide complete information to the stakeholders regarding all the technical and other restrictions, risk, or ambiguous requirements allowing them to take informed decisions. This information is provided in the form of a test report that includes details like what is missing or what is wrong with the system, to make stakeholders fully understand the issues affecting quality of the software product.

7.2 Levels of Tests for Testing Software

Before our software product is deployed or released, it has to undergo a testing process in which we will ensure that our health monitoring system is working in the manner in which it was intended. There are three main stages of testing that we have to complete before our system ready to release or deploy. These stages include unit testing, integration testing, and system testing.

7.2.1 Unit Testing

Unit testing is the first stage of testing in which we tested each component of the system in isolation. This is done to ensure that each individual component is performing its intended functionality. In other words, the objective of this stage is to determine whether the components of system are functioning as they were designed to. For this usually white box testing technique is preferred.

7.2.2 Integration testing

In this testing phase, we combine tested different modules or components of our software modules as a group in order to checks the data flow from one module to other modules. There are multiple ways through which we can test how different components of our system function at their interface, we can adopt either a top-down or bottom-up integration method.

7.2.3 System Testing

At last, in this stage of the testing we tested all the components of our system as a whole to ensure that the overall product meets the specified requirements. This phase is very important because it allows us to verify whether our application or system meets all the functional and business requirements that is demanded by our customer.

7.3 Test Management Process

In the test management process we have to manage all the testing activities to make sure that a high quality software product is ready to be delivered. In this process we have to organize and control the testing process. At the same time we also have to ensure the traceability and visibility of the testing process in order to deliver a high quality healthcare system. A proper testing plan is developed for carrying out all testing activities smoothly and efficiently. A test plan is helpful in defining the testing approach, its scope, required resources, and schedule of all the testing activities.

7.3.1 Design the Test Strategy

It is a very critical phase of the testing process where we have to define a set of guidelines or principles that determines the test design and regulate how we go about testing the product and achieve the goals. The main objective of the Test Strategy Phase is to provide an efficient approach to the testing process so that we are able to ensure quality, traceability, reliability and better planning.

Following are some important Sections of the Test Strategy Design:

a. Scope and overview:

First we have to define the scope of testing activities that need to be carried out with timelines w.r.t project timelines defined in the test plan.

b. The Test Approach

Then we have to define the level of testing that will be carried out on the product as well as the roles and responsibilities of every team member.

c. Testing Tools:

Then we have to define the tools required for test execution and for performance, load and security testing.

d. Risk Analysis

Then we have to enlist all possible risks that may occur during the testing. A risk mitigation strategy must also be defined to mitigate these risks and also a contingency plan in case some risks occurs.

7.3.2 Test Criteria

Test criteria is defined to help us (testers) to organize the test process. Test criteria is chosen in accordance with the available test effort. We also have to define test coverage measures (as a ratio between the test cases required) for satisfying the criteria. These test coverage measures will be used to obtain information regarding the completeness of integration tests. The aim is to enable us (testers) to specify integration tests and to evaluate the results in terms of test completeness.

7.3.3 Resource Planning

Table 7.1: Resource Planning

SR No.	Module	Tester Name	Tools	Time Required
1	Sign-Up/ Sign-In	Ehsaan Ali	MS Word, Visual Studio Code, Django Unit Testing	1 Hour
2	TBP	Daim Attiq	MS Word, Visual Studio Code, Django Unit Testing	1 Hour
3	EPI	Ehsaan Ali	MS Word, Visual Studio Code, Django Unit Testing	1 Hour
4	LHW	Daim Attiq	MS Word, Visual Studio Code, Django Unit Testing	1 Hour
5	LHW Evaluation	Ehsaan Ali	MS Word, Visual Studio Code, Django Unit Testing	1 Hour
6	Trends Analysis	Daim Attiq	MS Word, Visual Studio Code, Django Unit Testing	1 Hour
7	Reports	Ehsaan Ali	MS Word, Visual Studio Code, Django Unit Testing	1 Hour
8	Forecasting	Daim Attiq	MS Word, Visual Studio Code, Django Unit Testing	1 Hour
9	Records	Daim Attiq	MS Word, Visual Studio Code, Django Unit Testing	1 Hour

7.3.4 Plan Test Environment

We know that testing environment is a setup of hardware and software for the testers to efficiently execute test cases. But test environment can be configured as per the need of the system which is to be tested. Setting up a right test environment will ensure success in software testing phase. But we also have to be careful because any flaws or mistake in this process may lead to increase in cost and time to the client.

The test environment for our system will require setting up of various number of distinct areas such as:

a. Setup of Test Server

In order to test our health monitoring system we will be establishing a test server, which will support our application. pgAdmin which is a PostgreSQL based local server will be used for that purpose.

b. Network

We will also have to setup a network as per the test requirement. It will include a LAN Wifi setup or a private network setup

c. Test PC setup

We will also need a PC on which we will test our dashboard on different web browsers e.g. Chrome, Firefox, Opera etc.

d. Bug Reporting Tool

We will also have to define a tool that will be used to manage and report the bugs in the system. For our project we will be using GitHub's built in bug tracker. GitHub's tracker is called Issues, and has its own section in every repository.

7.3.5 Schedule and Estimation

Table 7.2: Schedule and Estimation

SR No.	Module	Tester Name	Start Date	End Date	Time Required
1	Sign-Up/ Sign-In	Ehsaan Ali	May-24-2021	May-24-2021	1 Hour
2	TBP	Daim Attiq	May-24-2021	May-24-2021	1 Hour
3	EPI	Ehsaan Ali	May-25-2021	May-25-2021	1 Hour
4	LHW	Daim Attiq	May-25-2021	May-25-2021	1 Hour
5	LHW Evaluation	Ehsaan Ali	May-26-2021	May-26-2021	1 Hour
6	Trends Analysis	Daim Attiq	May-26-2021	May-26-2021	1 Hour
7	Reports	Ehsaan Ali	June04-2021	June04-2021	1 Hour
8	Forecasting	Daim Attiq	June04-2021	June04-2021	1 Hour
9	Records	Daim Attiq	June04-2021	June04-2021	1 Hour

7.4 Test Cases With Description

Table 7.3: Test Cases With Description

Test Case ID	Test Case	Test Steps	Expected Output
1	User Login with correct details	1. User visits the dashboard 2. User clicks on "Sign-In" 3. User enter his username and password.	Login Successfully
2	User Login with incorrect username	1. User visits the dashboard 2. User clicks on "Sign-In". 3. User enter his username and password.	Login Failed- Incorrect Username
3	User Login with incorrect password	1. User visits the dashboard 2. User clicks on "Sign-In". 3. User enter his username and password.	Login Failed- Incorrect Password
4	User Login with incorrect username and password	1. User visits the dashboard 2. User clicks on "Sign-In". 3. User enter his username and password.	Login Failed- Incorrect Credentials
5	Add New User (With Correct Inputs)	1. Admin navigates to "Add New User" form. 2. Admin Enter details of new user. 3. Admin clicks on "Submit"	Registration Successful
6	Add New User (With Invalid Password Entry)	1. Admin navigates to "Add New User" form. 2. Admin Enter details of new user. 3. Admin clicks on "Submit"	Registration Failed – "Password must be minimum 8 characters.
7	Add New User(With Invalid Username Entry)	1. Admin navigates to "Add New User" form. 2. Admin Enter details of new user. 3. Admin clicks on "Submit"	Registration Failed – "Username must be combination of alphabets with numbers.
8	Add New User(With Invalid Password Entry)	1. Admin navigates to "Add New User" form. 2. Admin Enter details of new user. 3. Admin clicks on "Submit"	Registration Failed – "Password Not Match".
9	Add New User(With Invalid Email Entry)	1. Admin navigates to "Add New User" form. 2. Admin Enter details of new user. 3. Admin clicks on "Submit"	Registration Failed – "Invalid Email".
10	Monitor TBP Program	1. User login to the system. 2. User selects "TBP".	System Displays all Graphs and Charts related to TBP.
11	Monitor EPI Program	1. User login to the system. 2. User selects "EPI".	System Displays all Graphs and Charts related to EPI.
12	Monitor LHW Program	1. User login to the system. 2. User selects "LHWP".	System Displays all Graphs and Charts related to LHWP.
13	Download Graph (TBP)	1. User login to the system. 2. User selects "TBP". 3. System Displays graphs related to that program.	Graphs is downloaded successfully.
14	Download Graph (EPI)	1. User login to the system. 2. User selects "EPI". 3. System Displays graphs related to that program.	Graphs is downloaded successfully.
15	Download Graph (LHWP)	1. User login to the system. 2. User selects "LHWP". 3. System Displays graphs related to that program.	Graphs is downloaded successfully.
16	LHW Evaluation (Area Wise)	1. User login to the system. 2. User clicks "LHW Evaluation". 3. Then user selects "Area Wise"	System Displays all Graphs and Charts related to LHW Evaluation "Area Wise".

17	LHW Evaluation (Report Wise)	<ol style="list-style-type: none"> 1. User login to the system. 2. User clicks "LHW Evaluation". 3. Then user selects "Report Wise" 	System Displays all Graphs and Charts related to LHW Evaluation "Report Wise".
18	Trends Analysis	<ol style="list-style-type: none"> 1. User clicks on Trends Analysis" from the menu bar of the dashboard. 2. User selects the district. 3. User selects a time span and clicks on Generate". 	The system displays the current healthcare trends in the form of a graph.
19	Trends Analysis (Without Specifying Any District)	<ol style="list-style-type: none"> 1. User clicks on Trends Analysis" from the menu bar of the dashboard. 2. User selects a time span without selecting any district and clicks on Generate". 	The system should display an error message "Invalid Entry -Districted Not Selected"
20	Trends Analysis (Without Specifying Any Time Span)	<ol style="list-style-type: none"> 1. User clicks on Trends Analysis" from the menu bar of the dashboard. 2. User selects the district. 3. User does not selects a time span and clicks on Generate". 	The system should display an error message "Invalid Entry –Time Span Not Selected"
21	Forecasting (Area Covered By LHW) – With Correct Inputs	<ol style="list-style-type: none"> 1. User clicks on "Forecasting" from the menu bar of the dashboard. 2. User selects "Area Covered By LHW" 	The system should predict the total area that will be covered and display the results in the form of a graph.
22	Forecasting (Area Covered By LHW) – Without Specifying Any District	<ol style="list-style-type: none"> 1. User clicks on "Forecasting" from the menu bar of the dashboard. 2. User selects "Area Covered By LHW" 	The system should display an error message "Invalid Entry -Districted Not Selected"
23	Forecasting (Area Covered By LHW) – Without Specifying Total LHW	<ol style="list-style-type: none"> 1. User clicks on "Forecasting" from the menu bar of the dashboard. 2. User selects "Area Covered By LHW" 	The system should display an error message "Invalid Entry -LHW Not Entered"
24	Forecasting (Area Covered By LHW) – Without Specifying Submitted Reports	<ol style="list-style-type: none"> 3. User clicks on "Forecasting" from the menu bar of the dashboard. 4. User selects "Area Covered By LHW" 	The system should display an error message "Invalid Entry –No of Submitted Report Not Entered"
25	Generate Reports (With Correct Inputs)	<ol style="list-style-type: none"> 1. User clicks on "Reports" from the menu bar of the dashboard. 2. User selects the healthcare program and district. 3. User selects a time span and clicks on Generate". 	The system generates a report of the selected healthcare program.
26	Generate Reports (With Incorrect Inputs)	<ol style="list-style-type: none"> 1. User clicks on Trends Analysis" from the menu bar of the dashboard. 2. User selects a healthcare program. 3. User selects a time span without selecting any district and clicks on "Generate". 	The system should display an error message "Invalid Entry -Districted Not Selected"
27	Generate Reports (Without Specifying Any Time Span)	<ol style="list-style-type: none"> 1. User clicks on Trends Analysis" from the menu bar of the dashboard. 2. User selects a healthcare program. 3. User selects a district. 4. User does not selects a time span and clicks on "Generate". 	The system should display an error message "Invalid Entry –Time Span Not Selected"

7.4.0.1 Test Case Report

Table 7.4: Test Case Report

Test Case ID	Test Scenario	Test Steps	Input	Expected Output	Actual Output	Pass /Fail	Remarks
1	User Login with correct details	<ol style="list-style-type: none"> 1. User visits the dashboard 2. User clicks on "Sign-In" 3. User enter his username and password. 	Username = daim35 Password = Pakistan1947	Login Successfully	Login Successfully	Pass	As Expected
2	User Login with incorrect username	<ol style="list-style-type: none"> 1. User visits the dashboard 2. User clicks on "Sign-In". 3. User enter his username and password. 	Username = daim30 Password = Pakistan1947	Login Failed-Incorrect Username	Login Successfully	Fail	System should have given an error message indicating incorrect username.
3	User Login with incorrect password	<ol style="list-style-type: none"> 1. User visits the dashboard 2. User clicks on "Sign-In". 3. User enter his username and password. 	Username = daim35 Password = pak1947	Login Failed-Incorrect Password	Login Failed	Pass	As Expected
4	User Login with incorrect username and password	<ol style="list-style-type: none"> 1. User visits the dashboard 2. User clicks on "Sign-In". 3. User enter his username and password. 	Username = daim3 Password = pak1947	Login Failed-Incorrect Credentials	Login Failed	Pass	As Expected
5	Add New User (With Correct Inputs)	<ol style="list-style-type: none"> 1. Admin navigates to "Add New User" form. 2. Admin Enter details of new user. 3. Admin clicks on "Submit" 	First name = Daim, Last Name = Attiq, Username = daim45, Email = daim35@gmail.com , Password = Pak12345, Confirm Password = Pak12345	Registration Successful	Registration Successful	Pass	As Expected
6	Add New User (With Invalid Password Entry)	<ol style="list-style-type: none"> 1. Admin navigates to "Add New User" form. 2. Admin Enter details of new user. 3. Admin clicks on "Submit" 	First name = Daim, Last Name = Attiq, Username = daim45, Email = daim35@gmail.com	Registration Failed – "Password must be minimum 8	Registration Successful	Fail	New user must not be registered because password must be

			l.com , Password = Pak123, Confirm Password = Pak123	character s.			minimum 8 char.
7	Add New User(With Invalid Username Entry)	<ol style="list-style-type: none"> Admin navigates to "Add New User" form. Admin Enter details of new user. Admin clicks on "Submit" 	First name = Daim, Last Name = Attiq, Username = daim, Email = daim35@gmail.com , Password = Pak123, Confirm Password = Pak123	Registration Failed – "Username must be combination of alphabets with numbers.	Registration Failed	Pass	As Expected
8	Add New User(With Invalid Password Entry)	<ol style="list-style-type: none"> Admin navigates to "Add New User" form. Admin Enter details of new user. Admin clicks on "Submit" 	First name = Daim, Last Name = Attiq, Username = daim20, Email = daim35@gmail.com , Password = Pak12345, Confirm Password = Pak123	Registration Failed – "Password Not Match".	Registration Successful	Fail	New user must not be registered because password are not same. minimum 8 char.
9	Add New User(With Invalid Email Entry)	<ol style="list-style-type: none"> Admin navigates to "Add New User" form. Admin Enter details of new user. Admin clicks on "Submit" 	First name = Daim, Last Name = Attiq, Username = daim, Email = daim35@gmail.com , Password = Pak123, Confirm Password = Pak123	Registration Failed – "Invalid Email".	Registration Successful	Fail	New user must not be registered because E-mail is Invalid.
10	Monitor TBP Program	<ol style="list-style-type: none"> User login to the system. User selects "TBP". 	User selects "TBP" from the Menu Bar by clicking on it.	System Displays all Graphs and Charts related to TBP.	System Displays all Graphs and Charts related to TBP.	Pass	As Expected
11	Monitor EPI Program	<ol style="list-style-type: none"> User login to the system. User selects "EPI". 	User selects "EPI" from the Menu Bar by clicking on it.	System Displays all Graphs	System Displays all Graphs	Pass	As Expected

				and Charts related to EPI.	and Charts related to EPI.		
12	Monitor LHW Program	1. User login to the system. 2. User selects "LHWP".	User selects "LHWP" from the Menu Bar by clicking on it.	System Displays all Graphs and Charts related to LHWP.	System Displays all Graphs and Charts related to LHWP.	Pass	As Expected
13	Download Graph (TBP)	1. User login to the system. 2. User selects "TBP". 3. System Displays graphs related to that program.	User selects "TBP" from the Menu Bar by clicking on it. User clicks on "Download graphs". User Selects a format e.g. PNG, SVG	Graphs is downloa ded successfu lly.	Graphs is downloa ded successfu lly.	Pass	As Expected
14	Download Graph (EPI)	1. User login to the system. 2. User selects "EPI". 3. System Displays graphs related to that program.	User selects "EPI" from the Menu Bar by clicking on it. User clicks on "Download graphs". User Selects a format e.g. PNG, SVG	Graphs is downloa ded successfu lly.	Graphs is downloa ded successfu lly.	Pass	As Expected
15	Download Graph (LHWP)	1. User login to the system. 2. User selects "LHWP". 3. System Displays graphs related to that program.	User selects "LHWP" from the Menu Bar by clicking on it. User clicks on "Download graphs". User Selects a format e.g. PNG, SVG	Graphs is downloa ded successfu lly.	Graphs is downloa ded successfu lly.	Pass	As Expected
16	LHW Evaluation (Area Wise)	1. User login to the system. 2. User clicks "LHW Evaluation".	User selects "LHW Evaluation" from the Menu Bar by clicking on it.	System Displays all Graphs and Charts related	System Displays all Graphs and Charts related	Pass	As Expected

		3. Then user selects "Area Wise"	User clicks on "Area Wise".	to LHW Evaluation "Area Wise".	to LHW Evaluation "Area Wise".		
17	LHW Evaluation (Report Wise)	1. User login to the system. 2. User clicks "LHW Evaluation". 3. Then user selects "Report Wise"	User selects "LHW Evaluation" from the Menu Bar by clicking on it. User clicks on "Report Wise".	System Displays all Graphs and Charts related to LHW Evaluation "Report Wise".	System Displays all Graphs and Charts related to LHW Evaluation "Report Wise".	Pass	As Expected
18	Trends Analysis	1. User clicks on Trends Analysis" from the menu bar of the dashboard. 2. User selects the district. 3. User selects a time span and clicks on Generate".	District = "Jamshoro". Time Span = 2015 to 2020	The system displays the current healthcare trends in the form of a graph.	The system displays the current healthcare trends in the form of a graph.	Pass	As Expected
19	Trends Analysis (Without Specifying Any District)	1. User clicks on Trends Analysis" from the menu bar of the dashboard. 2. User selects a time span without selecting any district and clicks on Generate".	Time Span = 2015 to 2020	The system should display an error message "Invalid Entry - Districted Not Selected"	The system displays the current healthcare trends of previous district in the form of a graph.	Fail	The system should have display an error message because district was not selected.
20	Trends Analysis (Without Specifying Any Time Span)	1. User clicks on Trends Analysis" from the menu bar of the dashboard. 2. User selects the district. 3. User does not selects a time span and clicks on Generate".	District = "Badin".	The system should display an error message "Invalid Entry – Time Span Not Selected"	The system displayed an error message "Invalid Entry – Time Span Not Selected"	Pass	As Expected
21	Forecasting (Area Covered By LHW) – With	1. User clicks on "Forecasting" from the menu bar of the dashboard.	District = "Badin", Total LHW = "1500",	The system should predict the total area that	The system predicted the total area that	Pass	As Expected

	Correct Inputs	2. User selects "Area Covered By LHW"	Reports Submitted = "3500"	will be covered and display the results in the form of a graph.	will be covered and display the results in the form of a graph.		
22	Forecasting (Area Covered By LHW) – Without Specifying Any District	1. User clicks on "Forecasting" from the menu bar of the dashboard. 2. User selects "Area Covered By LHW"	District = "Empty", Total LHW = "1500", Reports Submitted = "3500"	The system should display an error message "Invalid Entry - Districted Not Selected"	The system displayed a blank page.	Fail	The system should display an error message "Invalid Entry - Districted Not Selected"
23	Forecasting (Area Covered By LHW) – Without Specifying Total LHW	1. User clicks on "Forecasting" from the menu bar of the dashboard. 2. User selects "Area Covered By LHW"	District = "Badin", Total LHW = "Empty", Reports Submitted = "3500"	The system should display an error message "Invalid Entry - LHW Not Entered"	The system displayed a blank page.	Fail	The system should have display an error message "Invalid Entry - LHW Not Entered"
24	Forecasting (Area Covered By LHW) – Without Specifying Submitted Reports	3. User clicks on "Forecasting" from the menu bar of the dashboard. 4. User selects "Area Covered By LHW"	District = "Badin", Total LHW = "3000", Reports Submitted = "Empty"	The system should display an error message "Invalid Entry – No of Submitted Report Not Entered"	The system displayed a blank page.	Fail	The system should have display an error message "Invalid Entry – No of Submitted Report Not Entered"
25	Generate Reports (With Correct Inputs)	1. User clicks on "Reports" from the menu bar of the dashboard. 2. User selects the healthcare program and district. 3. User selects a time span and clicks on "Generate".	Healthcare Program = "TBP" District = "Jamshoro". Time Span = 2015 to 2020	The system generates a report of the selected healthcare program.	The system generates a report of the selected healthcare program.	Pass	As Expected

26	Generate Reports (With Incorrect Inputs)	<ol style="list-style-type: none"> 1. User clicks on Trends Analysis" from the menu bar of the dashboard. 2. User selects a healthcare program. 3. User selects a time span without selecting any district and clicks on "Generate". 	Healthcare Program = "EPI" Time Span = 2015 to 2020	The system should display an error message "Invalid Entry - Districted Not Selected"	The system displayed a Blank Page	Fail	The system should have display an error message because district was not selected.
27	Generate Reports (Without Specifying Any Time Span)	<ol style="list-style-type: none"> 1. User clicks on Trends Analysis" from the menu bar of the dashboard. 2. User selects a healthcare program. 3. User selects a district. 4. User does not selects a time span and clicks on "Generate". 	Healthcare Program = "LHW" District = "Badin".	The system should display an error message "Invalid Entry – Time Span Not Selected"	The system displayed an error message "Invalid Entry – Time Span Not Selected"	Pass	As Expected

7.4.0.1 Test Findings: A total of 27 test cases were executed to test our system to check whether it conforms to the demanded specifications or not. Out of these 27 test cases 8 test cases were being resulted as “fail” and the remaining 19 test cases were being resulted as “Pass”. We tested each and every module with correct as well as incorrect inputs and found major issues were in “Report” section of the dashboard. When we tried to test the “Report” section with incorrect inputs, instead of displaying an error message it displayed a blank page which is against the specifications demanded by the stakeholder. Other than that same problem was found in the “Forecasting” section. Test cases were successfully executed with correct inputs but when incorrect inputs were given, the “Forecasting” section also displayed a blank page instead of displaying an error message.

7.5 Bugs Report

Table 7.5: Bug Report

Bug ID	Module Name	Bug Detail	Reporter	OS	Browser	Expected result	Actual Result	Severity
BG01	Add New User	System registers new user even if Passwords is<8 characters.	Ehsaan Ali	Windows 10.1	Chrome	Registration Failed	Registration Successful	High
BG02	Add New User	System registers new user even if "Password" and "Confirm Password" do no match.	Ehsaan Ali	Windows 10.1	Chrome	New user must not be registered because password are not same.	Registration Successful	High
BG03	Add New User	System registers new user even if E-mail is Invalid.	Ehsaan Ali	Windows 10.1	Chrome	New user must not be registered because E-mail is Invalid.	Registration Successful	High
BG04	Trends Analysis	System does not give error message indicating "No District Selected"	Daim Attiq	Windows 10.1	Chrome	The system should display an error message "Invalid Entry -Districted Not Selected"	The system displays the current healthcare trends of previous district in the form of a graph.	High
BG05	Forecasting	System does not give error message indicating "Enter Total LHW "	Daim Attiq	Windows 10.1	Chrome	The system should display an error message "Invalid Entry -LHW Not Entered"	The system displayed a blank page.	Medium
BG06	Forecasting	System does not give error message indicating "No of Submitted Report Not Entered"	Daim Attiq	Windows 10.1	Chrome	The system should display an error message ""No of Submitted Report Not Entered"	The system displayed a blank page.	Medium
BG07	Reports	System does not give error message indicating "Time Span Not Selected"	Ehsaan Ali	Windows 10.1	Chrome	The system should display an error message "Time Span Not Selected"	The system displayed a blank page.	Medium

Chapter 8

CONCLUSION

8.1 Conclusion

The aim behind developing a health monitoring dashboard is to allow intelligent monitoring of the healthcare programs and assist the healthcare sector in improving healthcare services. This system is also designed to assist in informed decision making by providing data insights and predicting future risks. The problem with the other existing healthcare monitoring dashboards is that they are limited because their User Interface (UI) is not interactive and data is not presented in an effective way. They also fail to provide the necessary information such as trend analysis and disease forecasting which is extremely useful in decision making. To overcome these limitations we have developed an intelligent dashboard that will utilize the data generated by the healthcare sector in improving healthcare services and tackle problems in overlooked health programs such as Tuberculosis program, Expanded Program of Immunization, and Lady Health Worker Program.

To develop the system we have followed “Scrum methodology” which is an agile framework for development of a software product. We have chosen it because it is flexible model and supports requirement change even after development is started. Moreover, our system also utilizes a large volume of data provided by the Sindh government related to the above-mentioned healthcare programs to perform disease forecasting using deep learning algorithm which will ultimately help in improving healthcare quality and decision making. While implementing deep learning, data preprocessing was performed to remove missing values, attribute selection will be done using domain knowledge and then model building evaluation will be performed. Linear Regression and Recurrent Neural Network (RNN) algorithms are used to predict the “Area that is expected to be covered by the Lady Health Workers” and also to predict the expected number of “Infant Deaths” in a district. Our product is intended to be used by decision/policy makers of Sindh to assist them in monitoring of the healthcare programs and understanding the healthcare trends. The forecasting capability of our system will help decision makers to prepare for future risks and threats in advance.

8.2 Future Work

Currently our system only monitors three major healthcare programs which include TB, EPI and LHWP because the scope of the project was limited to these programs only. But in the future it can be expanded to cover other healthcare programs e.g. National Maternal and Newborn Child Health (MNCH), HIV/AIDS Control Program, Malaria Control Program, and Food and Nutrition Program to effectively address the health problems faced in multiple regions of the country.

Our intelligent health monitoring dashboard is designed to forecast the area of a division or a district that is “Expected to be Covered by the Lady Health Workers” in that region based on the reports submitted by them. Other than that our system is also capable to predict the “Expected Infant Deaths” in a Division or District to allow the decision makers to prepare for these risks in advance. But in the future the forecasting capabilities of the system can be expanded to cover the TB program and predict the “Expected Deaths due to TB” and “No of expected TB Patients” in a region to effectively monitor and manage the TB Program.

References

- [1] J. Sun and C. K. Reddy, “Big data analytics for healthcare,” in Proceedings of the 19th ACM SIGKDD international conference on Knowledge discovery and data mining - KDD '13, 2013, p. 1525.
- [2] M. Shahbaz, C. Gao, L. L. Zhai, F. Shahzad, and Y. Hu, “Investigating the adoption of big data analytics in healthcare: the moderating role of resistance to change,” *J. Big Data*, vol. 6, no. 1, pp. 1–20, Dec. 2019.
- [3] “About — City Health Dashboard.” <https://www.cityhealthdashboard.com/about> (accessed Oct. 02, 2020).
- [4] “Public Health Dashboard - PHE.” <https://fingertips.phe.org.uk/public-health-dashboard-ftpage/0/gid/1938133143/pat/6/par/E12000004/ati/102/are/E06000015/cid/4/tbm/1/page-options/ovw-do-0> (accessed Oct. 02, 2020).
- [5] “About WHO.” <https://www.who.int/about> (accessed Oct. 02, 2020).
- [6] “Health.” <https://www150.statcan.gc.ca/n1/en/subjects/Health> (accessed Nov. 13, 2020).
- [7] “Health Dashboard.” <http://103.247.238.92/webportal/pages> (accessed Nov. 13, 2020).
- [8] S. Qazi, M. Usman, and A. Mahmood, “A data-driven framework for introducing predictive analytics into expanded program on immunization in Pakistan,” *Wien. Klin. Wochenschr.*, pp. 1–8, Sep. 2020.
- [9] M. Asad, A. Mahmood, and M. Usman, “A machine learning-based framework for Predicting Treatment Failure in tuberculosis: A case study of six countries,” *Tuberculosis (Edinb).*, vol. 123, p. 101944, Jul. 2020.
- [10] “What is Scrum Methodology and Scrum Project Management.” <https://digital.ai/resources/agile-101/what-is-scrum> (accessed Nov. 03, 2020).
- [11] A. Srivastava, S. Bhardwaj, and S. Saraswat, “SCRUM model for agile methodology,” in *Proceeding - IEEE International Conference on Computing, Communication and Automation, ICCCA 2017*, Dec. 2017, vol. 2017-January, pp. 864–869.
- [12] “Requirement Gathering Techniques - Tutorialspoint.” <https://www.tutorialspoint.com/business.analysis/requirement.gathering.techniques> (accessed Nov. 17, 2020).